



TUGAS AKHIR - TE 145561

**PENGUJIAN ARUS PADA MODUL PRAKTIKUM STARTING
MOTOR INDUKSI 3 FASA DENGAN METODE DOL, WYE-
DELTA, DAN SOFT STARTING**

Ghiok Nanda Alivsky
NRP 2214038022

Dosen Pembimbing I
Ir. Arif Musthofa, MT.

Dosen Pembimbing II
Suwito, ST., MT.

PROGRAM STUDI TEKNIK LISTRIK
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya
2017

-----*Halaman ini sengaja dikosongkan*-----



FINAL PROJECT - TE 145561

**CURRENT TESTING ON PRACTICUM MODULE STARTING A 3
PHASE INDUCTION MOTOR WITH DOL, WYE-DELTA, AND
SOFT STARTING METHODS**

Ghiok Nanda Alivsky
NRP 2214038022

Advisor I
Ir. Arif Musthofa, MT.

Advisor II
Suwito, ST., MT.

ELECTRICAL ENGINEERING STUDY PROGRAM
Electrical and Automation Engineering Department
Vocational Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2017

-----Halaman ini sengaja dikosongkan-----

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “**Pengujian Arus Pada Modul Praktikum Starting Motor Induksi 3 Fasa Dengan Metode DOL, Wye-Delta, Dan Soft Starting**” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, **20 Juli 2017**

-----Halaman ini sengaja dikosongkan-----

**PENGUJIAN ARUS PADA MODUL PRAKTIKUM
STARTING MOTOR INDUKSI 3 FASE DENGAN METODE
DOL, WYE-DELTA, DAN SOFT STARTING**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya**

Pada

**Program Studi Teknik Listrik
Departemen Teknik Elektro Otomasi
Fakultas Vokasi**

Institut Teknologi Sepuluh Nopember

Menyetujui:

Dosen Pembimbing I

Ir. Arif Musthofa MT

NIP. 196608 11199203 1004

Dosen Pembimbing II

Suwito ST. MT

NIP. 19810105 200501 1 004

**SURABAYA
JULI, 2017**

-----*Halaman ini sengaja dikosongkan*-----

PENGUJIAN ARUS PADA MODUL PRAKTIKUM STARTING MOTOR INDUKSI 3 FASA DENGAN METODE DOL, WYE- DELTA, DAN SOFT STARTING

Nama : Ghiok Nanda Alivsky
Pembimbing 1 : Ir.Arif Musthofa, MT.
Pembimbing 2: Suwito ST, MT

ABSTRAK

Motor induksi adalah alat listrik yang mengubah energi listrik menjadi energi mekanik. Motor Induksi mempunyai banyak keunggulan di segi teknis maupun ekonomis, karena itu motor induksi terutama jenis motor induksi 3 fasa banyak digunakan pada dunia industri. Motor induksi terutama jenis motor induksi 3 fasa memerlukan suatu metode untuk pengasutan motor. Metode pengasutan (*starting*) motor induksi 3 fasa yang akan dibahas antara lain metode DOL, *Wye-Delta*, dan *Soft Starting*. Akan tetapi motor induksi juga mempunyai kekurangan, antara lain arus awal yang besar yang dapat mengakibatkan penurunan tegangan sistem dan mengganggu kerja sistem peralatan lain dalam satu saluran. Arus awal *Soft Starting* akan dibandingkan dengan arus awal *starting* menggunakan metode DOL dan *Wye-Delta* dimana sistem perbandingan itu dilakukan dengan menggunakan satu modul yang dapat berfungsi untuk *menstart* motor dengan menggunakan ketiga metode tersebut secara bergantian dengan menampilkan nilai arus *realtime* dan arus lonjakan pada tampilan LCD. Hasil dari Modul *Starting Motor Induksi 3 Fasa* yaitu motor berputar menggunakan metode DOL dengan tegangan 380v, menggunakan metode *Wye-Delta* dengan dua kali tegangan input yaitu 380v untuk *Star* dan 220v untuk *Delta*, menggunakan metode *Soft Starting* tegangan input yang bertahap mulai 1% hingga 100% dari tegangan 380v *Delta*.

Kata Kunci : DOL (*Direct On-Line*), *Wye-Delta*, *Soft Starting*, *Starting Motor Induksi 3 Fasa*.

-----Halaman ini sengaja dikosongkan-----

CURRENT TESTING ON PRACTICUM MODULE STARTING A 3 PHASE INDUCTION MOTOR WITH DOL, WYE-DELTA, AND SOFT STARTING METHODS

Name : Ghiok Nanda Alivsky
Advisor 1 : Ir.Arif Musthofa, MT.
Advisor 2 : Suwito ST, MT

ABSTRACT

An induction motor is an electrical device that converts electrical energy into mechanical energy. Induction motor has many advantages in terms of technical and economical, therefore induction motors, especially types of 3-phase induction motors are widely used in the industrial world. An induction motor, especially a 3 phase induction motor type, requires a method for starting motor. Starting method of 3 phase induction motor that will be discussed include DOL, Wye-Delta, and Soft Starting methods. However, induction motors also have disadvantages, such as large initial currents that can lead to decreased system voltage and interfere with the working of other equipment systems in a single channel. The initial flow of Soft Starting will be compared with the starting current using the DOL and Wye-Delta methods where the comparison system is performed using a single module that can serve to start the motor by using the three methods in turn by displaying realtime current values and the surge current on the LCD display . Results from the Starting Module of 3 Phase Induction Motors is a rotating motor using the DOL method with a voltage of 380v, using the Wye-Delta method with twice the input voltage ie 380v for Star and 220v for Delta, using Soft Starting method of input voltage gradually from 1% to 100% of 380v Delta voltage.

Keywords : DOL (Direct On-Line), Wye-Delta, Soft Starting, Starting of 3 Phase Induction Motor.

-----Halaman ini sengaja dikosongkan-----

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT atas dua nikmat yang sangat berharga yaitu nikmat sehat dan juga nikmat waktu sehingga Tugas Akhir ini dapat terselesaikan dengan baik. Shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma-3 pada Bidang Studi Teknik Teknik Listrik, Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya dengan judul:

PENGUJIAN ARUS PADA MODUL PRAKTIKUM STARTING MOTOR INDUKSI 3 FASA DENGAN METODE DOL, WYE-DELTA, DAN SOFT STARTING

Penulis mengucapkan terima kasih kepada Ibu dan Bapak penulis yang memberikan berbagai bentuk doa serta dukungan tulus tiada henti, Bapak Ir. Arif Musthofa, MT. dan Bapak Suwito S.T., MT. atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya Tugas Akhir ini, Alma Quintha Devancha yang selalu memberikan doa, semangat, dan dukungannya kepada penulis. Penulis juga mengucapkan banyak terima kasih kepada teman teman kelas D3 PLN angkatan 2014 secara umumnya dan warga Kene secara khususnya, tentunya semua pihak yang telah membantu dalam proses penyelesaian Tugas Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, **20 Juli 2017**

Penulis

-----Halaman ini sengaja dikosongkan-----

DAFTAR ISI

	HALAMAN
HALAMAN JUDUL	i
PERNYATAAN KEASLIAN TUGAS AKHIR	v
HALAMAN PENGESAHAN	vii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
1. BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Batasan Masalah	1
1.4 Tujuan Perencanaan	2
1.5 Metodologi Penelitian	2
1.6 Sistematika Laporan Tugas Akhir	3
1.7 Relevansi	3
2. BAB II DASAR TEORI	5
2.1 Motor Listrik	5
2.2 Prinsip Kerja Motor Listrik	6
2.3 Motor Induksi Tiga Fasa	7
2.3.1 Klasifikasi Motor Induksi	8
2.3.2 Prinsip Kerja Motor Induksi 3 fasa	8
2.4 Metode-Metode <i>Starting</i> Motor Tiga Fasa	9
2.4.1 <i>Starting</i> dengan Metode <i>Direct On Line</i>	10
2.4.2 <i>Starting</i> dengan Metode <i>Autotransformer</i>	11
2.4.3 <i>Starting</i> dengan Metode <i>Star-Delta</i>	12
2.4.4 <i>Starting</i> dengan Metode <i>Soft Starter</i>	13
2.5 Komponen-Komponen Elektronika Daya	15
2.5.1 Dioda	15
2.5.2 Transistor	18
2.5.3 Thyristor	18
2.6 Mikrokontroler ATmega328	19
2.7 <i>Zero Crossing Detector</i>	22
2.8 MOC3021	23

3.	BAB III PERANCANGAN ALAT.....	25
3.1	Diagram Blok Keseluruhan Sistem.....	25
3.2	Prinsip Kerja Alat	26
3.3	Spesifikasi Motor Induksi 3 Fasa.....	26
3.4	Perancangan <i>Hardware</i> Sensor Arus	27
3.5	Perancangan Box Alat	30
3.5.1	Perancangan LCD.....	33
3.6	Perancangan <i>Interface</i> Pada Modul	34
4.	BAB IV MODUL PRAKTIKUM DAN PENGUJIAN SISTEM ..	37
4.1	Modul Praktikum Starting Motor Induksi 3 Fasa Untuk Percobaan Dalam Praktikum.....	37
4.2	Pengujian Sistem.....	48
4.2.1	Pengujian Sensor Arus CT-103	49
4.2.2	Pengujian Rangkaian Kontrol Dari Metode DOL (Direct On-Line), Wye-Delta, dan Soft Starting	53
4.2.3	Pengujian Motor Induksi 3 Fasa	57
4.2.4	Pengujian Keseluruhan.....	58
5.	BAB V PENUTUP	61
5.1	Kesimpulan	61
5.2	Saran	62
6.	DAFTAR PUSTAKA.....	63
7.	LAMPIRAN A.....	A-1
A.1.	Listing Program pada Arduino.....	A-1
8.	LAMPIRAN B.....	B-1
B.1	DATASHEET ARDUINO UNO	B-1
B.2.	DATASHEET PC817.....	B-5
B.3.	DATASHEET MOC3021	B-9
B.4	DATASHEET BT138	B-13
B.5	DATASHEET CT103	B-17
9.	LAMPIRAN C.....	C-1
C.1.	Tampilan Board.....	C-1
C.2.	Tampilan Depan Modul.....	C-2
C.3.	Tampilan Dalam Modul	C-2
10.	DAFTAR RIWAYAT HIDUP	D-1

DAFTAR GAMBAR

HALAMAN

Gambar 2.1 Motor Listrik.....	6
Gambar 2.2 Prinsip Kerja Motor Listrik.....	7
Gambar 2.3 Motor tipe rotor sangkar tupai (<i>Squirrel-cage rotor</i>)	8
Gambar 2.4 Diagram <i>Direct On Line Starter</i>	10
Gambar 2.5 Diagram <i>Autotransformer Starter</i>	12
Gambar 2.6 Diagram <i>Star-Delta Starter</i>	13
Gambar 2.7 Diagram <i>Soft Starter</i>	14
Gambar 2.8 Persambungan P-N Dioda.....	15
Gambar 2.9 Pembiasan Dioda	16
Gambar 2.10 Dioda <i>Bridge</i>	17
Gambar 2.11 Dioda Zener	17
Gambar 2.12 Simbol dan Gambar Transistor	18
Gambar 2.13 Simbol Dan Struktur Triac.....	19
Gambar 2.14 Mikrokontroler ATmega328 <i>Pin Mapping</i>	21
Gambar 2.15 Arduino Uno Dengan <i>Chip</i> ATmega328	21
Gambar 2.16 Sinyal <i>Zero Crossing Detector</i>	22
Gambar 2.17 Simbol MOC3021	23
Gambar 3.1 Diagram Blok Keseluruhan Sistem.....	25
Gambar 3.2 Motor Induksi 3 Fasa	27
Gambar 3.3 Perancangan <i>Hardware</i> Sensor Arus	28
Gambar 3.4 <i>Flowchart</i> Sensor Arus CT	29
Gambar 3.5 Foto <i>Box</i> Modul Tampak Depan.....	30
Gambar 3.6 Foto <i>Box</i> Modul Tampak Belakang	31
Gambar 3.7 Foto <i>Box</i> Modul Tampak Atas.....	31
Gambar 3.8 Desain <i>Box</i> Modul Tampak Atas	32
Gambar 3.9 Desain <i>Box</i> Modul Tampak Samping	32
Gambar 3.10 Desain <i>Box</i> Modul Tampak Depan	33
Gambar 3.11 Desain <i>Box</i> Modul Tampak Belakang	33
Gambar 3.12 Skematik Perancangan LCD	34
Gambar 3.13 <i>Interface</i> Pada Modul.....	35
Gambar 4.1 Rangkaian Daya Metode <i>DOL (Direct On-Line)</i>	38
Gambar 4.2 Rangkaian Kontrol Metode <i>DOL</i>	39
Gambar 4.3 Rangkaian Daya Metode <i>Wye-Delta</i>	42
Gambar 4.4 Rangkaian Kontrol Metode <i>Wye-Delta</i>	43

Gambar 4.5 Rangkaian Daya Metode <i>Soft Starting</i>	46
Gambar 4.6 Rangkaian Kontrol Metode <i>Soft Starting</i>	47
Gambar 4.7 Skema Pengujian Sensor CT	49
Gambar 4.8 Linierisasi Sensor CT 1 Fasa R Antara Clamp (sumbu Y) dengan Serial (sumbu X)	50
Gambar 4.9 Linierisasi Sensor CT 2 Fasa S Antara Clamp (sumbu Y) dengan Serial (sumbu X)	51
Gambar 4.10 Linierisasi Sensor CT 3 Fasa T Antara Clamp (sumbu Y) dengan Serial (sumbu X)	51
Gambar 4.11 Sampel Hasil Pengujian Sensor CT 1 (Fasa R) dengan Beban 25 watt	52
Gambar 4.12 Rangkaian Kontrol Metode DOL (<i>Direct On-Line</i>)	54
Gambar 4.13 Rangkaian Kontrol Metode <i>Wye-Delta</i>	54
Gambar 4.14 Rangkaian Kontrol Metode <i>Soft Starting</i>	55
Gambar 4.15 Rangkaian Kontrol pada Modul dengan Metode DOL..	55
Gambar 4.16 Rangkaian Kontrol pada Modul dengan Metode <i>Wye-Delta</i>	56
Gambar 4.17 Rangkaian Kontrol pada Modul dengan Metode <i>Soft Starting</i>	56
Gambar 4.18 Tampilan LCD setelah mendapat <i>Supply</i> Tegangan.....	57
Gambar 4.19 Pengujian Motor menggunakan Metode DOL	58
Gambar 4.20 Hasil pengukuran Arus fasa R, S, dan T pada Motor dengan metode DOL.....	58
Gambar 4.21 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode DOL (<i>Direct On-Line</i>) secara <i>Wye</i>	59
Gambar 4.22 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode DOL (<i>Direct On-Line</i>) secara <i>Delta</i>	59
Gambar 4.23 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode <i>Wye-Delta</i> saat <i>Wye</i>	60
Gambar 4.24 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode <i>Wye-Delta</i> saat <i>Delta</i>	60

DAFTAR TABEL

	HALAMAN
Tabel 4.1 Hasil Awal Pengujian Sensor CT	50
Tabel 4.2 Perbandingan Hasil Akhir dan Awal Serial Arduino UNO pada Pengujian Sensor CT	52
Tabel 4.3 Hasil pengujian motor induksi 3 fasa	57
Tabel 4.4 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode <i>Soft Starting</i>	60

-----Halaman ini sengaja dikosongkan-----

BAB I

PENDAHULUAN

1.1 Latar Belakang

Motor induksi mempunyai banyak keunggulan di segi teknis maupun ekonomis, karena itu motor induksi terutama jenis motor induksi 3 fasa terutama tipe sangkar tupai banyak digunakan pada dunia industri.

Saat motor induksi di starting secara langsung, arus starting motor besarnya 5 sampai 7 kali dari besar arus nominal. Ini akan menyebabkan drop tegangan yang besar pada jaringan listrik yang akan mengganggu perangkat-perangkat elektronik yang terkoneksi dengan jaringan listrik tersebut.

Sehingga diperlukan suatu metode pengasutan dengan pengurangan tegangan pada motor induksi yang bertujuan untuk mengurangi arus pengasutan awal. Metode pengasutan yang digunakan yaitu metode *soft starting*, diharapkan tegangan dan arus dari sumber tenaga dapat mengalir masuk kedalam motor AC secara bertahap, sehingga tidak memerlukan arus pengasutan yang besar dibandingkan dengan menggunakan metode DOL (*Direct On-Line*) dan *Wye-Delta*.

Dengan menggunakan ketiga metode tersebut, karakteristik arus starting akan dibandingkan. Dan diperlukan media pembelajaran sebagai salah satu cara untuk pembelajaran metode starting yang baik untuk motor induksi 3 fasa.

Dari bahasan paragraf diatas, maka kami realisasikan dengan menggunakan media modul metode starting motor induksi 3 fasa guna dapat digunakan untuk praktikum. Sehingga pengetahuan tentang metode starting motor induksi 3 fasa dapat berguna untuk pengaplikasian pada dunia industri.

1.2 Rumusan Masalah

Rumusan masalah yang akan dikaji dalam Tugas Akhir ini adalah bagaimana perbandingan karakteristik arus starting dari metode DOL (*Direct On-Line*), *Wye-Delta*, dan *Soft Starting* dengan modul praktikum *starting* motor induksi 3 fasa.

1.3 Batasan Masalah

Agar penulisan buku Tugas Akhir ini tidak menyimpang dan mengambang dari tujuan yang semula direncanakan sehingga

mempermudah mendapatkan data dan informasi yang diperlukan, maka penulis menetapkan batasan-batasan masalah sebagai berikut:

1. Analisa dibatasi pada respon motor induksi terhadap 3 metode starting yaitu DOL, Wye-Delta, dan Soft Starting.
2. Mengabaikan adanya harmonisa yang ditimbulkan pada masing-masing metode starting dan mengabaikan gelombang kurva arus dan tegangan.
3. Sensor arus yang digunakan yaitu CT-103 (Current Transformer).
4. Mikrokontroler yang digunakan adalah Aduino UNO (ATMega328) yang berfungsi untuk kalibrasi sensor arus CT.

1.4 Tujuan Perencanaan

Adapun tujuan dari Tugas Akhir ini yakni membuat suatu modul praktikum yang dikemas secara rapih yang terdiri dari tiga macam metode starting motor induksi 3 fasa dan mempunyai sistem interface guna memonitor hasil nilai arus starting yang akan dibandingkan pada setiap metodenya. Dan nantinya modul ini dapat digunakan sebagai media praktikum pada laboraturium Teknik Elektro Otomasi.

1.5 Metodologi Penelitian

Penelitian ini dilakukan melalui beberapa tahapan metodologi, yaitu studi literatur, permodelan sistem, perancangan dan pembuatan alat, analisa data dan penyusunan laporan buku Tugas Akhir.

Pada tahap studi literatur akan dipelajari tentang perangkat dan karakteristik arus dari macam-macam metode starting motor induksi 3 fasa, mempelajari macam macam komponen dari elektronika daya yang digunakan pada rangkaian DOL (Direct On-Line), Wye-Delta, dan Soft Starting. Pada tahap permodelan sistem, dirancang permodelan atau desain dari tata letak komponen dan wiring rangkaian DOL (Direct On-Line), Wye-Delta, dan Soft Starting. Kemudian perancangan hardware berupa desain tata letak setiap komponen dan wiring pada setiap metode starting motor induksi 3 fasa. Yang terakhir adalah merancang hasil permodelan sistem dan pengecekan terhadap sistem secara keseluruhan sehingga diperoleh data pengukuran yang akurat. Data percobaan yang diperoleh selanjutnya akan dianalisa. Dari hasil analisa, akan ditarik kesimpulan dari penelitian yang telah dilakukan. Tahapan akhir penelitian adalah penyusunan laporan penelitian.

1.6 Sistematika Laporan Tugas Akhir

Sistematika pembahasan Tugas Akhir ini terdiri dari lima bab, yaitu pendahuluan, teori penunjang, perencanaan dan pembuatan alat, pengujian dan analisa alat, serta penutup.

Bab I : PENDAHULUAN

Pada bab ini membahas tentang latar belakang permasalahan, permasalahan, tujuan, sistematika laporan, serta relevansi.

Bab II : DASAR TEORI

Pada bab ini membahas tentang teori penunjang yang mendukung dalam perencanaan pembuatan alat meliputi definisi dan prinsip kerja motor induksi tiga fasa, teori macam macam pengasutan motor induksi tiga fasa, teori definisi dari komponen komponen elektronika daya yang digunakan pada masing-masing metode *starting* motor induksi 3 fasa

Bab III : PERENCANAAN DAN PEMBUATAN ALAT

Membahas tentang perencanaan dan pembuatan perangkat keras (*hardware*) yang meliputi perancangan *software* sensor arus.

Bab IV : MODUL PRAKTIKUM DAN PENGUJIAN SISTEM

Membahas tentang pengukuran, pengujian, dan penganalisaan terhadap lonjakan arus *starting* pada metode *starting* motor induksi 3 fasa menggunakan prosedur penggunaan alat. Kemudian membahas tentang hasil analisa dan akan dibandingkan untuk masing-masing metode *starting* motor induksi 3 fasa.

Bab V : PENUTUP

Menjelaskan tentang kesimpulan dari tugas akhir ini dan saran-saran untuk pengembangan alat ini lebih lanjut.

1.7 Relevansi

Dengan adanya tugas akhir ini diharapkan dapat dijadikan sebagai tambahan materi Praktikum pada laboratorium D3 Teknik Elektro ITS dan juga salah satu solusi media pembelajaran bagi mahasiswa atau pelajar supaya dapat lebih memahami tentang beberapa macam *starting* motor 3 fasa, sehingga kedepannya dapat diaplikasikan pada dunia industri.

-----Halaman ini sengaja dikosongkan-----

BAB II

DASAR TEORI

Pada bab ini membahas `tentang teori dasar dan teori penunjang dari peralatan-peralatan yang digunakan dalam pembuatan Tugas Akhir “Pengujian Karakteristik Arus Starting Motor Induksi 3 Fasa Menggunakan Metode DOL, Wye-Delta, dan Soft Starting”.

2.1 Motor Listrik

Motor listrik adalah alat untuk mengubah energi listrik menjadi energi mekanik. Alat yang berfungsi sebaliknya, mengubah energi mekanik menjadi energi listrik disebut generator atau dinamo. Motor listrik dapat ditemukan pada peralatan rumah tangga seperti kipas angin, mesin cuci, pompa air dan penyedot debu. Motor listrik yang umum digunakan di dunia Industri adalah motor listrik *asinkron*, dengan dua standar global yakni IEC dan NEMA. Motor *asinkron* IEC berbasis *metrik* (milimeter), sedangkan motor listrik NEMA berbasis *imperial* (inch), dalam aplikasi ada satuan daya dalam *horsepower* (hp) maupun *kiloWatt* (kW).

Motor induksi adalah salah satu jenis dari motor listrik, yang dimana motor induksi tersebut merupakan motor yang paling umum digunakan pada berbagai peralatan industri. Popularitasnya karena rancangannya yang sederhana, murah dan mudah didapat, dan dapat langsung disambungkan ke sumber daya AC. Motor induksi memiliki dua komponen listrik utama yaitu:

1. Rotor. Motor induksi menggunakan dua jenis rotor: Rotor kandang tupai terdiri dari batang penghantar yang tebal diletakkan dalam petak-petak *slots* paralel. Batang-batang tersebut diberi hubungan pendek pada kedua ujungnya dengan alat cincin hubungan pendek. Lingkaran rotor yang memiliki gulungan tiga fase, lapisan ganda dan terdistribusi. Dibuat melingkar sebanyak kutub stator. Tiga fase digulungi kawat pada bagian dalamnya dan ujung yang lainnya dihubungkan ke cincin kecil yang dipasang pada batang as dengan sikat yang menempel padanya.

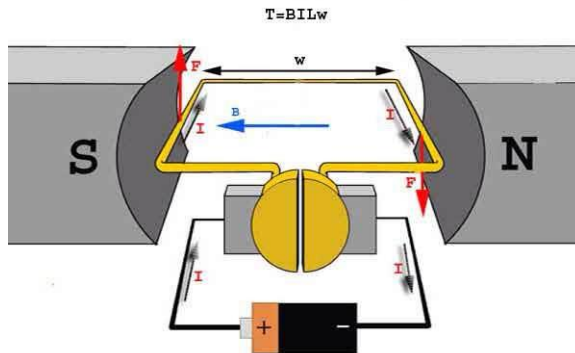
2. Stator. Stator dibuat dari sejumlah *stampings* dengan *slots* untuk membawa gulungan tiga fase. Gulungan ini dilingkarkan untuk sejumlah kutub yang tertentu. Gulungan diberi spasi geometri sebesar 120 derajat. (Daryanto, 2016:98)



Gambar 2.1 Motor Listrik

2.2 Prinsip Kerja Motor Listrik

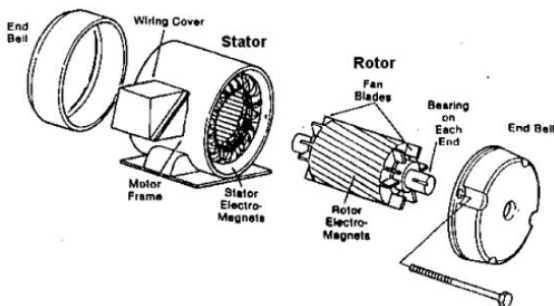
Pada motor listrik tenaga listrik diubah menjadi tenaga mekanik. Perubahan ini dilakukan dengan mengubah tenaga listrik menjadi magnet yang disebut sebagai elektro magnet. Sebagaimana kita ketahui bahwa : kutub-kutub dari magnet yang senama akan tolak-menolak dan kutub-kutub tidak senama, tarik-menarik. Maka kita dapat memperoleh gerakan jika kita menempatkan sebuah magnet pada sebuah poros yang dapat berputar, dan magnet yang lain pada suatu kedudukan yang tetap. Dapat kita ketahui dari Gambar 2.2.



Gambar 2.2 Prinsip Kerja Motor Listrik

2.3 Motor Induksi Tiga Fasa

Motor induksi tiga fasa memiliki dua komponen dasar yaitu stator dan rotor, bagian rotor dipisahkan dengan bagian stator oleh celah udara (*air gap*). Tipe dari motor induksi berdasarkan pada jenis rotor dibagi menjadi dua macam yaitu *Wound* rotor (tipe motor yang memiliki rotor terbuat dari lilitan) dan *Squirrel-cage* rotor (tipe motor induksi dimana konstruksi rotor disusun oleh beberapa batangan logam yang dimasukkan melewati slot-slot yang ada pada rotor motor induksi, kemudian pada setiap bagian disatukan oleh cincin yang membuat batangan logam dihubungkan singkat dengan batangan logam yang lain).



Gambar 2.3 Motor Tipe Rotor Sangkar Tupai (*Squirrel-Cage* Rotor)

2.3.1 Klasifikasi Motor Induksi

Motor induksi dapat diklasifikasikan menjadi dua kelompok utama (Parekh, 2003):

1. Motor induksi satu fase. Motor ini hanya memiliki satu gulungan stator, beroperasi dengan pasokan daya satu fase, memiliki sebuah rotor kandang tupai, dan memerlukan sebuah alat untuk menghidupkan motornya. Sejauh ini motor ini merupakan jenis motor yang paling umum digunakan dalam peralatan rumah tangga, seperti fan angin, mesin cuci dan pengering pakaian, dan untuk penggunaan hingga 3 sampai 4 Hp.
2. Motor induksi tiga fase. Medan magnet yang berputar dihasilkan oleh pasokan tiga fase yang seimbang. Motor tersebut memiliki kemampuan daya yang tinggi, dapat memiliki kandang tupai atau gulungan rotor (walaupun 90% memiliki rotor kandang tupai); dan penyalan sendiri. Diperkirakan bahwa sekitar 70% motor di industri menggunakan jenis ini, sebagai contoh, pompa, kompresor, *belt conveyor*, jaringan listrik, dan *grinder*. Tersedia dalam ukuran 1/3 hingga ratusan Hp.
(Daryanto, 2016:99)

2.3.2 Prinsip Kerja Motor Induksi 3 fasa

Motor induksi merupakan motor arus bolak-balik (AC) yang paling luas digunakan. Penamaannya berasal dari kenyataan bahwa arus rotor motor ini bukan diperoleh dari sumber tertentu, tetapi merupakan arus yang terinduksi sebagai akibat adanya perbedaan relatif antara putaran rotor dengan medan putar (*Rotating Magnetic field*) yang dihasilkan oleh arus stator.

Apabila ketiga belitan stator diberikan masing-masing diberi tegangan dari sumber tiga fasa, maka akan timbul medan magnet yang berputar dengan kecepatan sinkron.

$$n_s = \frac{120f}{p} \quad (2.1)$$

dengan:

n_s = Kecepatan medan putar stator (rpm)

f = Frekuensi jaringan (Hz)

p = Jumlah kutub

Medan putar tersebut akan memotong konduktor rotor hingga terbangkit tegangan induksi. Karena konduktor rotor dihubung singkat, maka akan mengalir arus dalam konduktor rotor. Arus rotor ini berada dalam medan magnet dari stator menurut hukum *Lorenz* akibatnya timbul gaya/torka. Bila gaya ini cukup untuk mengerakkan rotor maka ia akan berputar dengan kecepatan

$$n_r = (1 - s) n_s \quad (2.2)$$

dengan:

n_r = Kecepatan putar rotor (rpm)

s = Slip putaran

Karena tegangan induksi hanya akan terbangkitkan jika terjadi perpotongan antar medan putar dengan konduktor rotor maka kecepatan rotor tidak dapat menyamai kecepatan medan putar stator, harus ada selisih dimana kecepatan rotor (n_r) harus lebih rendah dari kecepatan medan putar (kecepatan sinkron n_s). Perbedaan kecepatan ini disebut *slip* (s) dan dinyatakan dengan:

$$s = \frac{n_s - n_r}{n_s} \times 100\% \quad (2.3)$$

adanya perbedaan kecepatan medan putar dan rotor ini sehingga mesinnya disebut mesin tak sinkron/serempak.

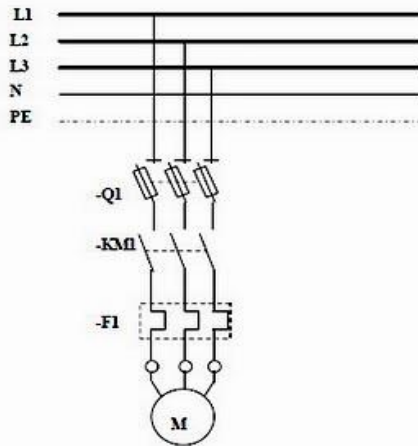
2.4 Metode-Metode *Starting* Motor Tiga Fasa

Pada motor induksi yang diam apabila tegangan normal diberikan ke stator, maka akan ditarik arus yang besar oleh belitan primernya. Motor induksi saat dihubung langsung akan menarik arus 5 sampai 7 kali dari arus beban penuh dan hanya menghasilkan torsi 1,5 dan 2,5 kali

torsi beban penuh. Arus mula yang besar ini dapat mengakibatkan drop tegangan pada saluran, sehingga akan mengganggu pada peralatan lain yang dihubungkan pada saluran yang sama. Untuk mengurangi besarnya arus *starting* pada motor induksi, ada beberapa cara atau metode starting yang biasa digunakan.

2.4.1 Starting dengan Metode *Direct On Line*

Direct On Line merupakan *starting* langsung. Penggunaan metoda ini sering dilakukan untuk motor-motor AC yang mempunyai kapasitas daya yang kecil. Pengertian penyambungan langsung disini, motor yang akan dijalankan langsung di *switch* on ke sumber tegangan jala-jala sesuai dengan besar tegangan nominal motor. Artinya tidak perlu mengatur atau menurunkan tegangan pada saat *starting*, arus *starting* sama dengan arus hubung singkat. Rangkaian *starting* DOL ini dapat dilihat pada Gambar 2.4



Gambar 2.4 Diagram *Direct On Line* Starter

Starter ini terdiri dari *Breaker* sebagai proteksi hubung singkat, *Magnetik Contactor*, *Over Current Relay* dan komponen control seperti *push button*, *MCB* dan *pilot lamp*. Kontrol Start dan Stop

dilakukan dengan *push button* yang mengontrol tegangan pada *coil contactor*. Sementara itu output OCR terangkai secara seri sehingga jika OCR trip, maka output OCR akan melepas tegangan ke *coil contactor*. Komponen penyusun starter ini harus mempunyai *ampacity* yang cukup besar. Perlu diperhitungkan juga arus saat start motor, demikian juga ukuran *range overloadnya*. Keuntungan pengasutan DOL adalah sebagai berikut :

1. Pengasutannya sederhana
2. Jarang salah menjalankannya
3. Instalasinya ekonomis
4. Efisiensi kerjanya tinggi

Kerugian pengasutan DOL adalah sebagai berikut

1. Tegangan sumber menurun tiba-tiba
2. Arus asut besar mengakibatkan kopel besar
3. Bila pengasutan sering dilakukan akan membangkitkan panas yang tinggi.

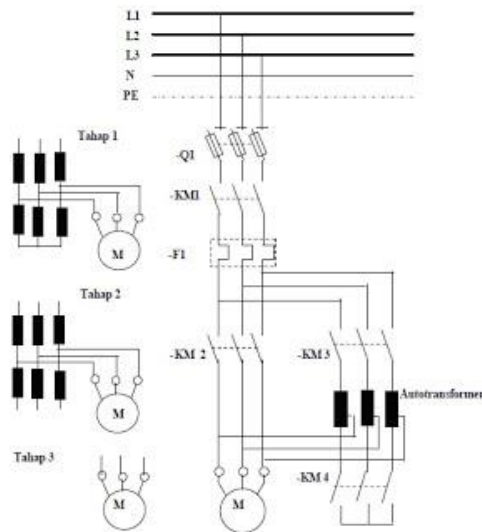
2.4.2 Starting dengan Metode Autotransformer

Pengasutan tipe ini mempunyai keuntungan utama yaitu diperoleh harga momen per-unit yang besar dari arus pengasutannya. Besar arus motor berkurang sebanding dengan tegangan terminal motor. Sedangkan arus jala-jala berkurang sebanding dengan arus kuadrat dari tegangan terminal motor, karena pengaruh dari autotrafo. Tetapi biaya pengasutan dengan metode ini umumnya lebih tinggi dibandingkan dengan metode pengasutan lainnya. Autotrafo umumnya dilengkapi dengan dua atau tiga tap, misalnya 80 %, 60 % dan 50 % tap tegangan. Pengasutan dengan autotrafo mengakibatkan ketidakseimbangan tegangan antara fasa. Kumparan trafo hanya digunakan pada saat pengasutan dan setelah itu dapat dilepaskan dari rangkaian. Pelepasan kumparan trafo ini dapat dilakukan secara otomatis dengan menggunakan kontak-kontak rele induksi dan dilengkapi dengan rele waktu. Dibawah ini ditunjukkan gambar pengasutan dengan autotrafo yang menggunakan dua kumparan trafo. Keuntungan pengasutan auto trafo:

1. Reaktansi bocor yang lebih rendah

2. Arus yang dihasilkan lebih kecil
3. Rugi-rugi yang lebih rendah

Kerugian pengasutan auto trafo adalah Langsungnya hubungan listrik antara sisi bertegangan tinggi dan rendah. Diagram starting *autotransformer* dapat dilihat pada Gambar 2.5

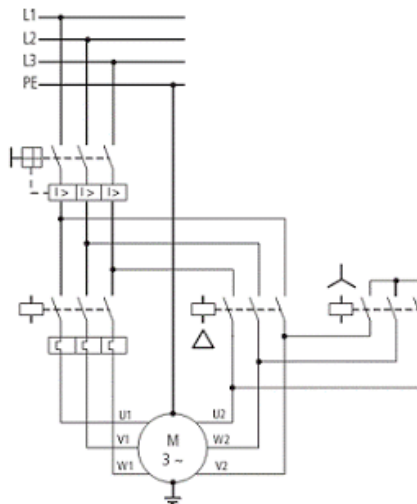


Gambar 2.5 Diagram *Autotransformer Starter*

2.4.3 Starting dengan Metode *Star-Delta*

Pengasutan dengan metode ini digunakan terutama untuk motor induksi rotor sangkar yang direncanakan bekerja/berputar pada hubungan delta. Ujung-ujung kumparan fasanya harus dikeluarkan ke terminal motor, yaitu untuk melakukan hubungan *Wye* dan hubungan *Delta*. Pada tipe ini mula-mula kumparan dihubungkan secara *Wye*, dimana tegangan fasanya adalah sebesar 0.577 dari tegangan kumparan pada saat dihubungkan *delta*, sedangkan impedensi kumparan tersebut bila dihubungkan *delta*. Dengan demikian, arus pengasutan pada

hubungan bintang adalah lebih kecil bila dihubungkan dengan *delta*. Demikian juga momen pengasutannya adalah sebanding dengan besarnya arus pengasutan yang timbul. Umumnya alat pengasutan ini dilengkapi dengan kontaktor-kontaktor yang dimiliki oleh masing-masing hubungan tersebut. Perpindahan hubungan *Wye* ke *Delta* didalam sistem pengasutan ini diatur oleh suatu rele pengatur waktu (*Time Delay Relay*). Pada saat perpindahan hubungan *Wye* ke *Delta* terjadi pemutusan arus sesaat dari arus jala-jala. Namun hal ini dapat diatasi dengan menggunakan sistem peralihan tertutup dengan hubungan sebuah tahanan ke dalam rangkaian selama perpindahan berlangsung. Diagram starting *star-delta* dapat dilihat pada Gambar 2.6



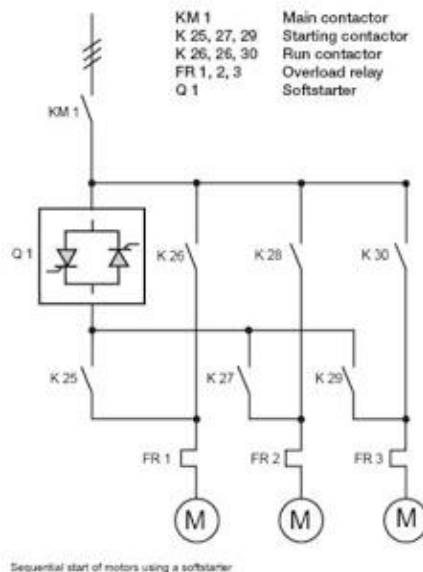
Gambar 2.6 Diagram *Star-Delta Starter*

2.4.4 Starting dengan Metode *Soft Starter*

Soft starter dipergunakan untuk mengatur/ memperhalus start dari elektrik motor. Prinsip kerjanya adalah dengan mengatur tegangan yang masuk ke motor. Pertama-tama motor hanya diberikan tegangan yang rendah sehingga arus dan torsi pun juga rendah. Pada level ini motor

hanya sekedar bergerak perlahan dan tidak menimbulkan kejutan. Selanjutnya tegangan akan dinaikan secara bertahap sampai ke nominal tegangannya dan motor akan berputar dengan dengan kondisi RPM yang nominal.

Komponen utama softstarter adalah thyristor dan rangkaian yang mengatur *trigger thyristor*. Seperti diketahui, *output thyristor* dapat di atur via pin *gate* nya. Rangkaian tersebut akan mengontrol level tegangan yang akan dikeluarkan oleh *thyristor*. *Thyristor* yang terpasang bisa pada 2 phase atau 3 phase. Diagram rangkaian metode *soft starter* dapat dilihat pada Gambar.2.7



Gambar 2.7 Diagram *Soft Starter*

2.5 Komponen-Komponen Elektronika Daya

Dalam dunia elektronika, selain kita mengenal komponen elektronika aktif dan pasif, juga terdapat salah satu jenis komponen lainnya, yakni komponen elektronika daya. Pasti sebagian dari anda banyak yang bertanya-tanya sebenarnya apa itu elektronika daya.

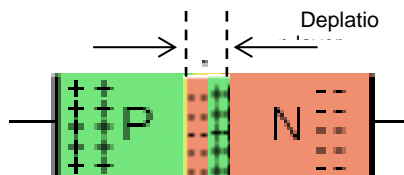
Perlu diketahui bahwa sebenarnya elektronika daya adalah cabang ilmu elektronika yang membahas terkait switching, pengontrolan serta pengubah atau konversi blok blok yang besar dari daya listrik menggunakan sarana alat berbahan semikonduktor. Secara garis besar, elektronika terbagi menjadi dua macam, yakni rangkaian daya serta rangkaian kontrol.

2.5.1 Dioda

Dioda merupakan komponen elektronika yang hanya bisa mengalirkan arus listrik satu arah saja, yaitu dari anoda ke katoda, hal ini disebabkan adanya potensial penghalang (*Potential Barrier*) pada persambungan PN. Material pembentuk dioda adalah semi konduktor dari bahan silikon atau germanium tipe P dan N, seperti terlihat pada Gambar 2.8.



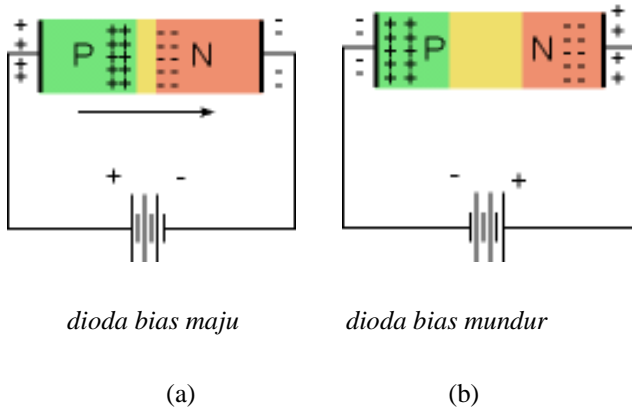
(a) Simbol dioda



(b) Konstruksi dioda

Gambar 2.8 Persambungan P-N Dioda

Struktur dalam dioda adalah sambungan semikonduktor tipe P dan N, yang mengakibatkan munculnya lapisan deplesi pada persambungannya. Dengan struktur demikian arus hanya akan dapat mengalir dari sisi P menuju sisi N, jika pada anoda dan katodanya diberikan tegangan maju seperti pada Gambar 2.9 a, Jika anoda dan katodanya diberikan tegangan mundur seperti pada Gambar 2.9 b, maka tidak terjadi aliran arus pada dioda. Sifat khusus dari dioda inilah yang kemudian melandasi terbentuknya konsep rangkaian penyearah arus AC menjadi DC.



Gambar 2.9 Pembiasan Dioda

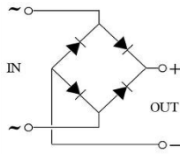
Ada beberapa jenis dioda seperti pada berikut :

1. Dioda Generic

Dioda generic yang paling umum digunakan sebagai penyearah tegangan AC dari jala-jala listrik ke tegangan DC yang siap dikonsumsi untuk berbagai keperluan catu daya. Proses penyearahan AC ke DC

2. Dioda Bridge

Secara bahan, fungsi dan karakteristik dioda bridge sama dengan dioda generic, bedanya hanya pada dalam satu kemasan dioda bridge tersusun dari empat dioda generic, seperti terlihat pada gambar 3.6.



(a) Simbol

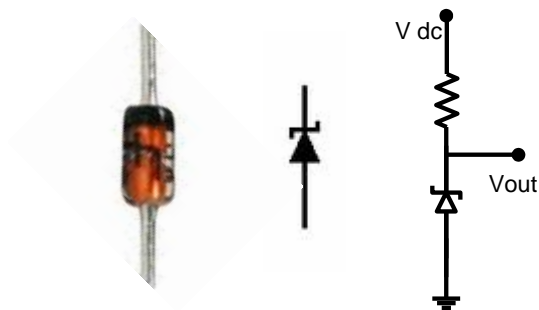


(b) Jenis

Gambar 2.10 Dioda Bridge

3. Dioda Zener

Dioda zener merupakan jenis dioda yang memanfaatkan kondisi *breakdown*, sehingga pemasangan dioda ini dalam rangkaian, dipasang secara *reverse* untuk mendapatkan titik PIV (*Peak Inverse Voltage*) yang telah ditentukan oleh pabrik. Dengan adanya PIV pada dioda zener ini, maka dimungkinkan pembatasan level tegangan yang stabil dalam sebuah sistem rangkaian elektronika . Contoh rangkaian dapat dilihat pada Gambar 2.11.

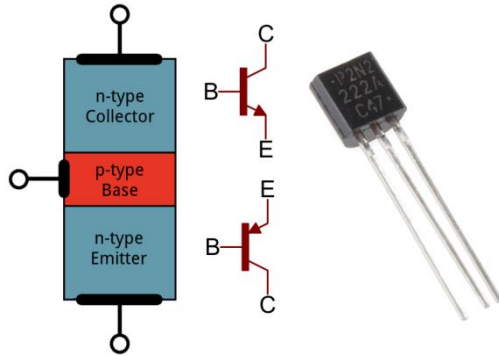


(a) Rangkaian dasar dioda zener

Gambar 2.11. Dioda Zener

2.5.2 Transistor

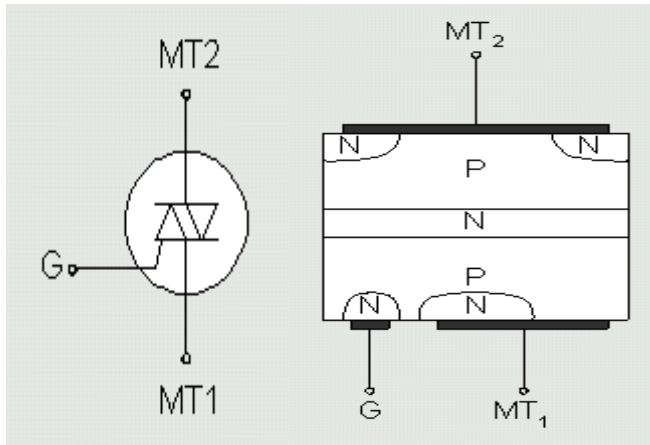
Transistor adalah komponen elektronika dasar yang juga masuk ke dalam jajaran komponen elektronika aktif. Komponen yang memiliki fungsi sebagai penguat dan switch ini juga termasuk salah satu jenis komponen elektronika daya yang sering dijumpai di berbagai jenis rangkaian elektronika.



Gambar 2.12. Simbol dan Gambar Transistor

2.5.3 Thyristor

Thyristor berakar kata dari bahasa Yunani yang berarti ‘pintu’. Dinamakan demikian karena sifat dari komponen yang mirip dengan pintu yang dapat dibuka dan ditutup untuk melewatkan arus listrik. Ciri - ciri utama dari sebuah thyristor adalah komponen yang terbuat dari bahan *semiconductor silicon*. Thyristor adalah komponen semikonduktor untuk pensaklaran yang berdasarkan pada struktur PNPN. Komponen ini memiliki kestabilan dalam dua keadaan yaitu on dan off serta memiliki umpan-balik regenerasi internal. Thyristor memiliki kemampuan untuk mensaklar arus searah (DC) yaitu jenis SCR, maupun arus bolak-balik (AC), jenis TRIAC. contoh thyristor yaitu triac pada Gambar 2.13



Gambar 2.13. Simbol Dan Struktur Triac

2.6 Mikrokontroler ATmega328

Mikrokontroler ATmega328 atau pada jaman sekarang bisa disebut dengan Arduino Uno secara sederhana, adalah chip yang menjadi otak dari sebuah rangkaian elektronika. Analoginya, jika pada komputer anda otak utama adalah *chip processor* Intel Core i7 (atau AMD) yang mengatur semua *hardware* dari mulai *motherboard*, RAM, *hardisk*, *monitor*, *keyboard*, dsb, maka pada rangkaian mikrokontroler Arduino, chip ATmega328 adalah otaknya, yang mengatur komponen lain yang terhubung dengan nya (misal led, sensor, motor). Jika pada komputer and *software* diinstall dengan *Operating System Windows* (atau Mac OSX / Linux) sebagai tuan rumah nya, pada mikrokontroler *software* berbentuk program yang langsung di 'bakar' (*burn*) langsung pada *chip* mikrokontroler yang bersangkutan, dengan '*bootloader*' sebagai tuan rumahnya. Pada platform Arduino sekarang, chip yang populer digunakan adalah ATmega328. Platform Arduino, yang dibantu oleh dengan board-board Arduino, memudahkan anda melakukan

memrogram chip ATmega328 untuk melakukan tugas yang anda tentukan.

ATmega328 adalah chip mikrokontroler 8-bit berbasis AVR-RISC buatan Atmel. *Chip* ini memiliki 32 KB memori ISP flash dengan kemampuan baca-tulis (*read write*), 1 KB EEPROM, dan 2 KB SRAM. Dari kapasitas memori Flash nya yang sebesar 32 KB itulah chip ini diberi nama ATmega328. Chip lain yang memiliki memori 8 KB diberi nama ATmega8, dan ATmega16 untuk yang memiliki memori 16 KB.

Chip ATmega328 memiliki banyak fasilitas dan kemewahan untuk sebuah chip mikrokontroler. Chip tersebut memiliki 23 jalur general purpose I/O (input/output), 32 buah register, 3 buah timer/counter dengan mode perbandingan, interrupt internal dan external, serial programmable USART, 2-wire interface serial, serial port SPI, 6 buah channel 10-bit A/D converter, programmable watchdog timer dengan oscillator internal, dan lima power saving mode. Chip bekerja pada tegangan antara 1.8V ~ 5.5V. Output komputasi bisa mencapai 1 MIPS per Mhz. *Maximum operating frequency* adalah 20 Mhz.

ATmega328 menjadi cukup populer setelah chip ini dipergunakan dalam board Arduino. Dengan adanya Arduino yang didukung oleh *software* Arduino IDE, pemrograman *chip* ATmega328 menjadi jauh lebih sederhana dan mudah.

Pada dasarnya *board* Arduino adalah tentang ATmega328. Board Arduino didesain untuk mempermudah kita untuk melakukan pemrograman dan koneksi *chip* ATmega328 dengan komponen lainnya. Dengan *board* Arduino, baik itu Arduino UNO, Arduino Mega 2560, Arduino Nano, maupun Arduino Pro Mini, semua nya membantu untuk menyederhanakan proses membuat rangkaian mikrokontroler. Contoh *pin mapping* dari mikrokontroler pada Gambar 2.14 dan gambar arduino uno pada Gambar 2.15

Arduino Pin Mapping

www.arduino.cc

digital pin 0 (RX)	(RESET) PC6	1	28	PC5 (ADC5/SCL)	analog input 5
digital pin 1 (TX)	(RXD) PD0	2	27	PC4 (ADC4/SDA)	analog input 4
digital pin 2	(TXD) PD1	3	26	PC3 (ADC3)	analog input 3
digital pin 3	(INT0) PD2	4	25	PC2 (ADC2)	analog input 2
digital pin 4	(INT1) PD3	5	24	PC1 (ADC1)	analog input 1
	(XCK/T0) PD4	6	23	PC0 (ADC0)	analog input 0
	VCC	7	22	GND	
	GND	8	21	AREF	
	(XTAL1/TOSC1) PB6	9	20	AVCC	
	(XTAL2/TOSC2) PB7	10	19	PB5 (SCK)	digital pin 13 (LED)
digital pin 5	(T1) PD5	11	18	PB4 (MISO)	digital pin 12
digital pin 6	(AIN0) PD6	12	17	PB3 (MOSI/OC2)	digital pin 11 (PWM)
digital pin 7	(AIN1) PD7	13	16	PB2 (SS/OC1B)	digital pin 10 (PWM)
digital pin 8	(ICP1) PB0	14	15	PB1 (OC1A)	digital pin 9 (PWM)

Gambar 2.14 Mikrokontroler ATmega328 *Pin Mapping*

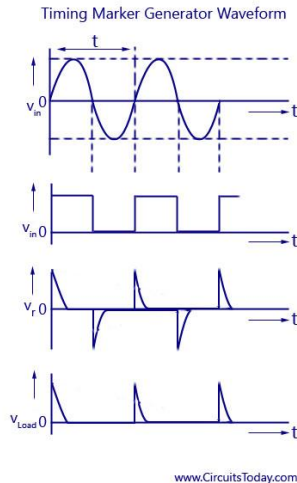


Gambar 2.15 Arduino Uno Dengan *Chip* ATmega328

2.7 Zero Crossing Detector

Zero crossing detector adalah rangkaian yang digunakan untuk mendeteksi gelombang sinus AC 220 volt saat melewati titik tegangan nol. Seberangan titik nol yang dideteksi adalah peralihan dari positif menuju negatif dan peralihan dari negatif menuju positif. Seberangan-seberangan titik nol ini merupakan acuan yang digunakan sebagai awal pemberian nilai waktu tunda untuk pemicuan *triac*.

Metode ini berfungsi untuk menentukan frekuensi suatu gelombang dengan cara mendeteksi banyaknya zero point pada suatu rentang waktu. *Zero crossing detector* berfungsi untuk mendeteksi perpotongan gelombang sinus pada tegangan AC dengan *zero point* tegangan AC tersebut, sehingga dapat memberikan sinyal acuan saat dimulainya pemicuan sinyal PWM. Dengan menggunakan rangkaian *zero crossing detector* ini, kita dapat mendeteksi *zero point* sekaligus mengubah suatu sinyal sinusoidal (*sine wave*) menjadi sinyal kotak (*square wave*). Perpotongan titik nol yang terdeteksi adalah pada saat peralihan dari siklus positif menuju siklus negatif dan peralihan dari siklus negatif menuju siklus positif. Contoh sinyal ada pada Gambar 2.16

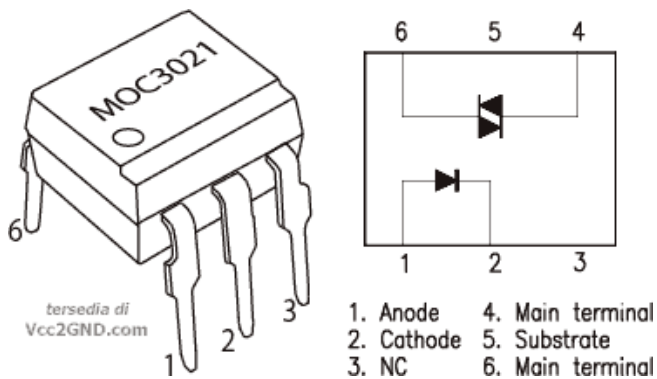


Gambar 2.16 Sinyal *Zero Crossing Detector*

2.8 MOC3021

MOC3021 adalah Optocoupler / Optoisolator TRIAC Driver yang dirancang untuk mengarah-kendalikan rangkaian elektronika bertegangan 220 Volt AC (listrik PLN) dalam kemasan standar DIP 6-pin.

Aplikasinya meliputi pengendali katup solenoid, *ballast* lampu, sebagai antarmuka antara MCU / rangkaian bertegangan rendah searah (*low voltage DC circuit*) dengan peralatan arus bolak-balik (*alternating current*) bertegangan hingga 240 Volt AC, pengarah-kendali motor elektrik (*AC motor driver*), sebagai pengatur intensitas cahaya lampu bohlam (*incandescent lamp dimmer*), dsb. Komponen elektronika saklar ganda (*bilateral switch*) ini menawarkan isolasi hingga 7500 Volt (kondisi puncak). Sehingga dengan komponen ini dapat melindungi mikrokontroler dari tegangan 220 volt AC. Contoh MOC3021 pada Gambar 2.17



Gambar 2.17 Simbol MOC3021

-----*Halaman ini sengaja dikosongkan*-----

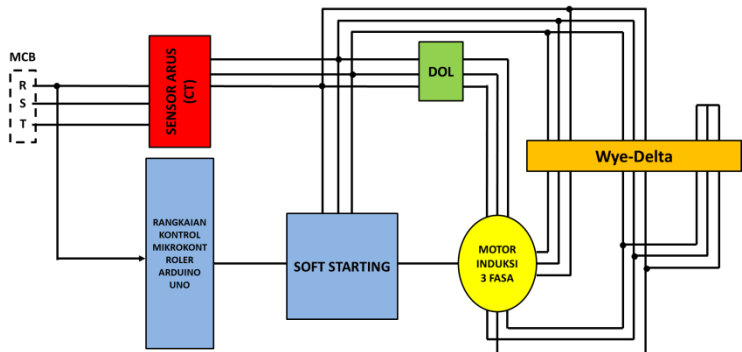
BAB III PERANCANGAN ALAT

Pada bab ini dibahas mengenai perancangan “Perancangan Modul Praktikum *Starting* Motor Induksi 3 Fasa”, baik perancangan perangkat keras (*hardware*) serta perangkat lunak (*software*) untuk sensor arus yang meliputi :

- a. Menentukan spesifikasi motor induksi 3 fasa
- b. Diagram blok sistem dari alat yang akan dirancang
- c. Perancangan Hardware, terdiri dari :
 1. Perancangan *wiring* metode DOL (*Direct On-Line*) ,
 2. Perancangan *wiring* metode *Wye-Delta*, dan
 3. Perancangan *wiring* metode *Soft Starting*
- d. Perancangan Software Sensor Arus
- e. Perancangan Box alat

3.1 Diagram blok keseluruhan

Perancangan perangkat keras (*hardware*) pada modul *starting* motor induksi 3 fasa meliputi sensor arus, rangkaian DOL (*Direct On-Line*), rangkaian *Wye-Delta*, dan rangkaian *Soft Starting*. Gambar diagram blok dari seluruh sistem rangkaian modul *starting* motor induksi 3 fasa dapat dilihat pada Gambar 3.2.



Gambar 3.1 Diagram Blok Keseluruhan Sistem

Rangkaian *starting* motor induksi 3 fasa yang terdiri dari 3 metode (DOL, *Wye-Delta*, *Soft Starting*) dirancang melewati sensor arus (*Current Transformer*) guna untuk mendeteksi besar arus *starting* pada masing-masing metode *starting* yang nantinya akan di tampilkan pada *interface* (LCD).

3.2 Prinsip Kerja Alat

Prinsip kerja alat dibedakan menjadi 3 sesuai dengan metode masing-masing rangkaian yaitu :

1. Prinsip kerja dari metode DOL (*Direct On-Line*) ini adalah memutar atau menghidupkan Motor listrik 3 fasa dengan satu arah putaran saja tanpa ada loncatan tegangan seperti rangkaian *Wye* atau *Delta*.
2. Prinsip kerja dari metode *Wye-Delta* ini adalah Ketika tombol start di tekan sesaat maka kondisi star akan aktif, sesaat kemudian kondisi star akan mati bergantian dengan kondisi delta yang akan menyala. Perpindahan antara kondisi star ke delta di atur waktunya secara otomatis oleh timer pada mikrokontroler. Sedangkan tombol stop berfungsi untuk memutus aliran listrik keseluruhan rangkaian.
3. Prinsip kerja dari metode *Soft Starting* ini adalah dengan mengatur tegangan yang akan masuk ke motor. Pertama-tama motor akan diberi tegangan yang rendah sehingga arus dan torsiya rendah. Pada kejadian ini motor hanya bergerak perlahan dan tidak menimbulkan kejutan. Selanjutnya tegangan dinaikan tahap demi tahap sampai pada tegangan nominalnya dan motor akan berputar dengan kondisi putaran yang nominal. Pada perencanaannya waktu yang dibutuhkan untuk proses soft starter ini yaitu dibutuhkan waktu satu detik untuk motor bisa bekerja pada keadaan nominalnya

3.3 Spesifikasi Motor Induksi 3 Fasa

Spesifikasi motor berfungsi untuk menunjukkan rating kerja motor itu sendiri. Adapun spesifikasi motor induksi 3 fasa yang kami gunakan sebagai berikut.

1. Daya : 0.25 kW

2. Frekuensi : 50 Hz
3. D koppling : 380 Volt
4. $I_{starting}$: 1.4 A
5. $I_{steady state}$: 0.8 A
6. Kecepatan putar : 1340 rpm

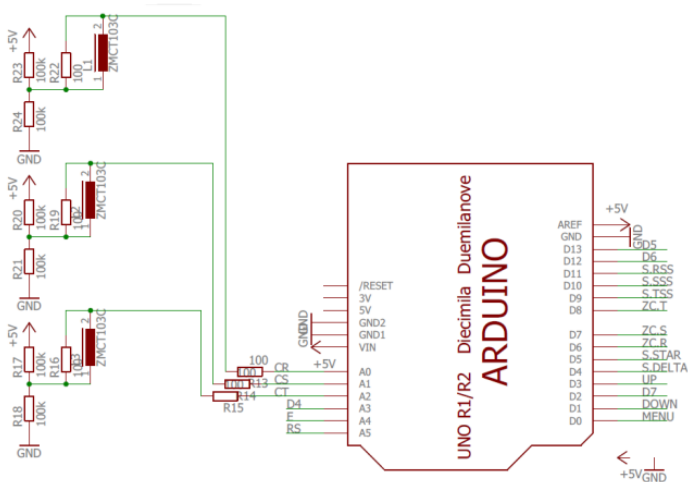
Sepesifikasi yang telah tersebutkan diatas, mempunyai bentuk *hardware* seperti yang ditunjukkan pada gambar 3.1.



Gambar 3.2 Motor Induksi 3 Fasa

3.4 Perancangan *Hardware* Sensor Arus

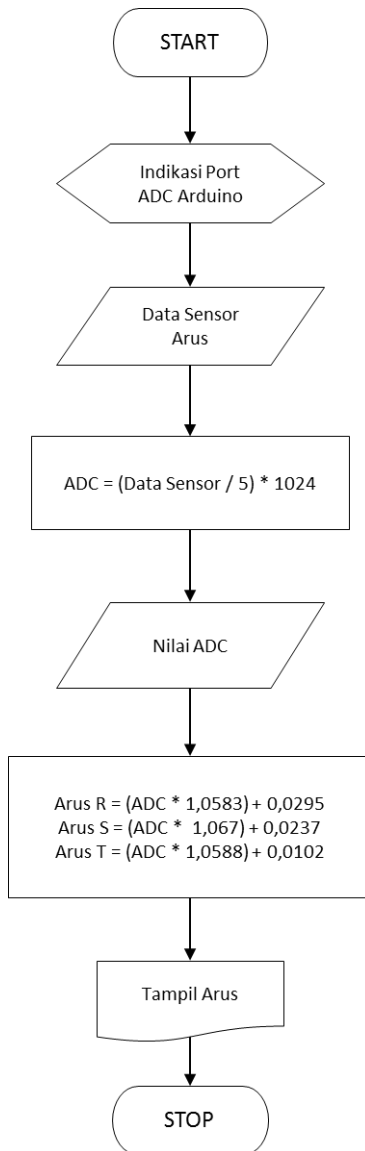
Perancangan *hardware* sensor arus ini terdiri dari kumpulan beberapa komponen yang membentuk suatu fungsi guna untuk mengukur arus pada rangkaian. Jenis sensor arus (*Current Transformer*) yang digunakan adalah ZMCT103C. Gambar rancangan *hardware* bisa dilihat pada Gambar 3.3.



Gambar 3.3 Perancangan *Hardware* Sensor Arus

3.4 Perancangan Software Sensor Arus

Perancangan perangkat lunak pada sensor arus yaitu berfungsi untuk memberikan instruksi pada perangkat untuk mendeteksi besar arus yang diukur. Untuk mempermudah dalam melakukan pemahaman tentang cara kerja sensor arus (*Current Transformer*), terlebih dahulu dibuat diagram alir (*flowchart*) yang dapat dilihat pada Gambar 3.4 dan program sensor arus CT pada Mikokontroler Aduino UNO (ATMega328) yang telah terlampir.

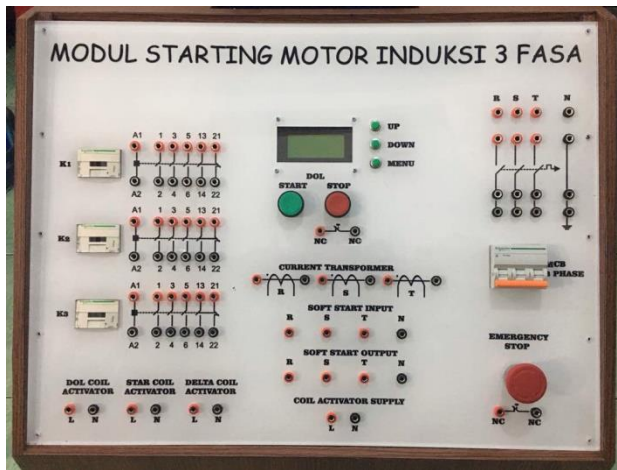


Gambar 3.4 *Flowchart* Sensor Arus CT

Setelah *flowchart* dibuat, kami akan melakukan kalibrasi pada sensor arus CT yang bertujuan untuk memperbaiki tingkat keakuratan dalam mendeteksi arus yang nantinya akan ditampilkan pada LCD. Sensor arus CT juga kami rancang mempunyai kemampuan untuk *record* arus lonjakan pada setiap metode yang akan dijalankan dengan cara mengambil data arus *peak to peak* atau puncak arus tertinggi.

3.5 Perancangan Box Alat

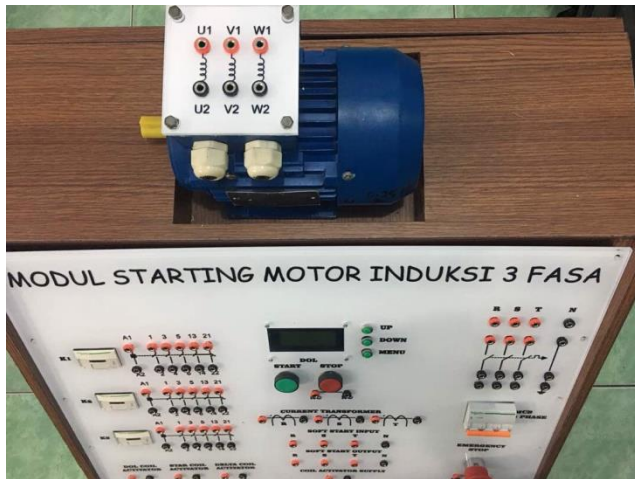
Adapun perancangan *box* atau *packaging* guna mempercantik tampilan dari alat ini serta melindungi komponen-komponen yang telah menjadi suatu sistem dari objek-objek yang dapat membahayakan pada komponen alat ini. Berikut adalah desain tata letak komponen pada box. Tampilan *box/packaging* dapat dilihat pada Gambar 3.5, Gambar 3.6, dan Gambar 3.7.



Gambar 3.5 Foto *Box* Modul Tampak Depan

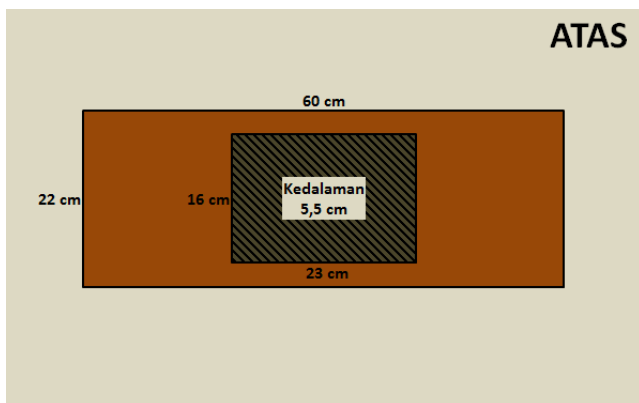


Gambar 3.6 Foto *Box* Modul Tampak Belakang

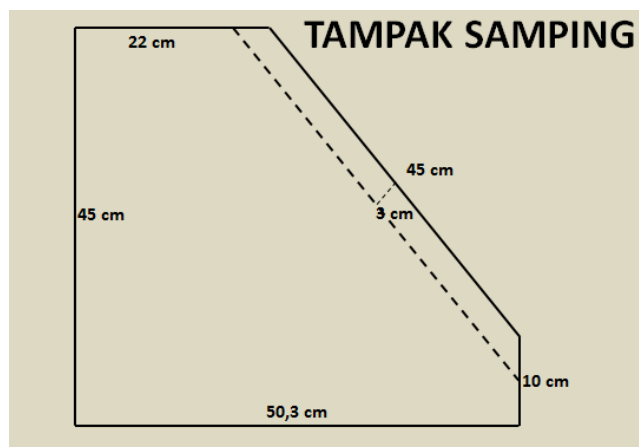


Gambar 3.7 Foto *Box* Modul Tampak Atas

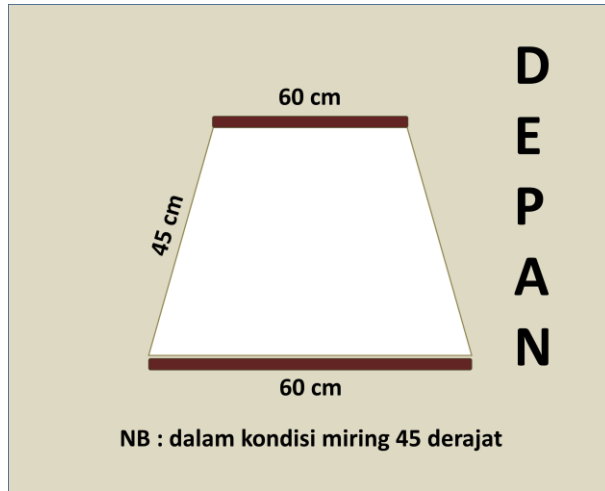
Adapun rancangan untuk dimensi *Box* Modul Starting Motor Induksi 3 Fasa yang dapat dilihat pada Gambar 3.8, Gambar 3.9, Gambar 3.10, dan Gambar 3.11.



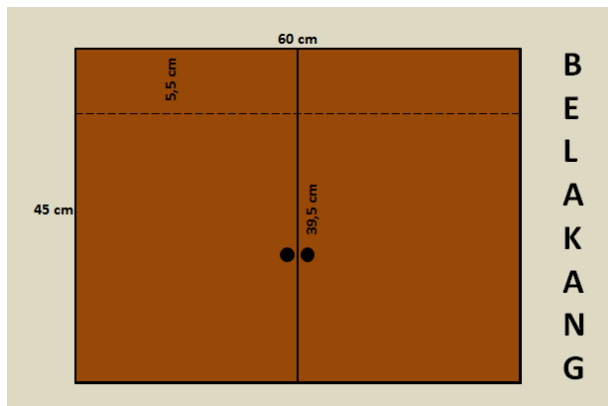
Gambar 3.8 Desain *Box* Modul Tampak Atas



Gambar 3.9 Desain *Box* Modul Tampak Samping



Gambar 3.10 Desain *Box* Modul Tampak Depan

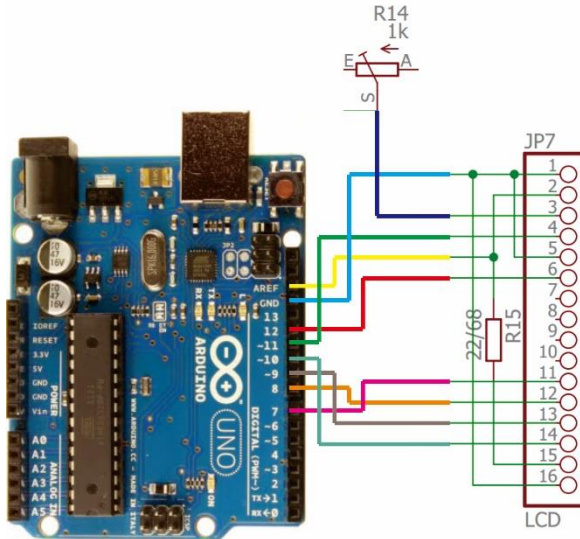


Gambar 3.11 Desain *Box* Modul Tampak Belakang

3.5.1 Perancangan LCD

Pada tugas akhir ini, tampilan akhir dari sensor arus ditampilkan pada sebuah LCD 20x4 tipe HD44780. Pin ground dan RW terhubung dengan pin ground pada Modul Arduino. Pin enable terhubung dengan

pin D12, pin RS terhubung dengan pin D11, pin D4, D5, D6 dan D7 masing-masing terhubung dengan pin D7, D8, D9 dan D10 dari Modul Arduino. Skematik dari LCD tersebut seperti yang ditampilkan pada Gambar 3.12.



Gambar 3.12 Skematik Perancangan LCD

3.6 Perancangan Interface Pada Modul

Sarana atau medium atau sistem operasi yang digunakan untuk menghubungkan antara perangkat lunak (*software*) agar dapat berkomunikasi dengan pengguna. *Interface* pada modul ini berfungsi untuk mengatur panjang pendeknya waktu *starting*, dalam arti lain dapat mengatur waktu mulai tidak ada tegangan sampai mencapai kondisi *steady state* dengan cara mengatur tegangan yang keluar sehingga dapat mengurangi arus *starting* yang dihasilkan oleh metode *Soft Starting*. *Interface* pada modul ini terdiri dari 1 buah LCD 20x4, tombol UP, tombol DOWN, dan tombol MENU Tampilan *interface* pada modul ini bisa dilihat pada Gambar 3.13.



Gambar 3.13 *Interface* Pada Modul

-----*Halaman ini sengaja dikosongkan*-----

BAB IV

MODUL PRAKTIKUM DAN PENGUJIAN SISTEM

Pada bab ini akan membahas mengenai modul yang digunakan untuk praktikum dan pengujian alat untuk “Starting Motor Induksi 3 Fasa dengan Metode DOL (*Direct On-Line*), *Wye-Delta*, dan *Soft Starting*. Data pengujian sangat diperlukan untuk implementasi dalam dunia nyata. Pengujian ini juga berguna untuk mengetahui apakah alat yang telah dibuat sudah berfungsi sebagaimana mestinya.

4.1 Modul Praktikum *Starting Motor Induksi 3 Fasa Untuk Percobaan Dalam Praktikum*

Pada sub bab ini menjelaskan tentang prosedur dan tata cara penggunaan modul Praktikum *Starting Motor Induksi 3 Fasa* dalam percobaan praktikum yang dimana prosedur ini digunakan untuk melakukan praktikum pada laboratorium Teknik Elektro Otomasi. Berikut adalah modul percobaan praktikum *Starting Motor Induksi 3 Fasa*.

MODUL PERCOBAAN PRAKTIKUM STARTING MOTOR INDUKSI 3 FASA

PERCOBAAN 1

STARTING MOTOR INDUKSI 3 FASA DENGAN MENGUNAKAN METODE DOL (Direct On-Line)

I. Tujuan :

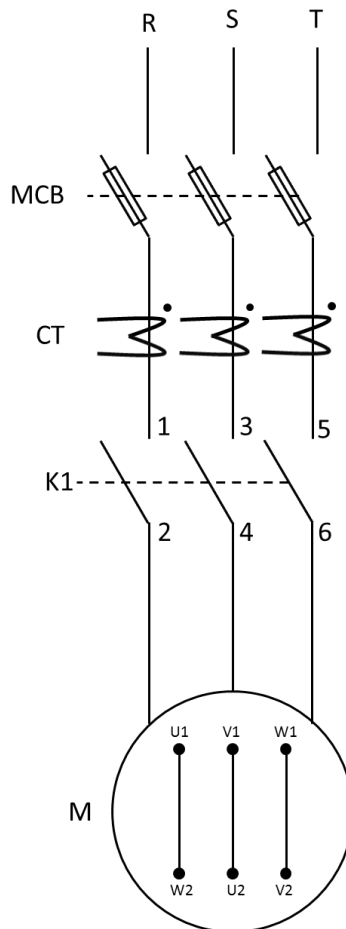
1. Memahami prinsip kerja starting motor induksi 3 fasa dengan menggunakan metode DOL (*Direct On-Line*).
2. Memahami karakteristik arus *starting* metode DOL (*Direct On-Line*).
3. Mengenal rangkaian daya dan rangkaian kontrol dari metode DOL (*Direct On-Line*).

II. Peralatan yang dibutuhkan:

1. Alas kaki
2. Modul percobaan
3. Kabel jumper

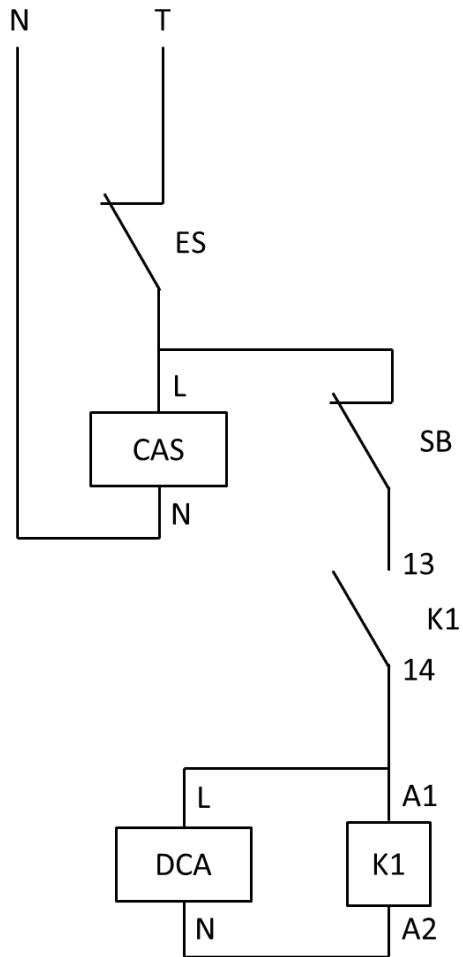
4. Motor induksi 3 fasa
5. Modul 'Starting Motor Induksi 3 Fasa'

III. Gambar rangkaian: DAYA



Gambar 4.1 Rangkaian Daya Metode *DOL* (*Direct On-Line*)

KONTROL



Gambar 4.2 Rangkaian Kontrol Metode *DOL*.

KETERANGAN

MCB	: Miniature Circuit Breaker
ES	: Emergency Switch
CAS	: Coil Activator Supply
DCA	: DOL Coil Activator
CT	: Current Transformer
SB	: Stop Button
M	: Motor

IV. Langkah percobaan:

1. Patuhi standar keamanan yang telah ditetapkan pada praktikum.
2. Pastikan sumber tegangan dalam keadaan mati.
3. Rangkailah rangkaian seperti pada gambar 4.1.
4. Rangkailah rangkaian kontrol DOL seperti pada gambar 4.2.
5. Sebelum menekan tombol 'START', laporkan pada asisten terlebih dahulu untuk melakukan pengecekan.
6. Pastikan *Emergency Switch* dalam keadaan NC (*Normal Close*).
7. Nyalakan MCB.
8. Tekan tombol 'MENU' untuk masuk ke pilihan metode yang akan digunakan.
9. Pilih menu DOL *pada interface* dengan menekan tombol 'Menu'.
10. Tekan tombol 'START' dan motor akan berputar.
11. Amati lalu catat angka arus yang ditampilkan oleh LCD.
12. Tekan tombol 'STOP' untuk menghentikan motor.
13. Matikan MCB.
14. Analisa hasil arus starting metode DOL.

PERCOBAAN 2
STARTING MOTOR INDUKSI 3 FASA DENGAN
MENGGUNAKAN METODE WYE-DELTA

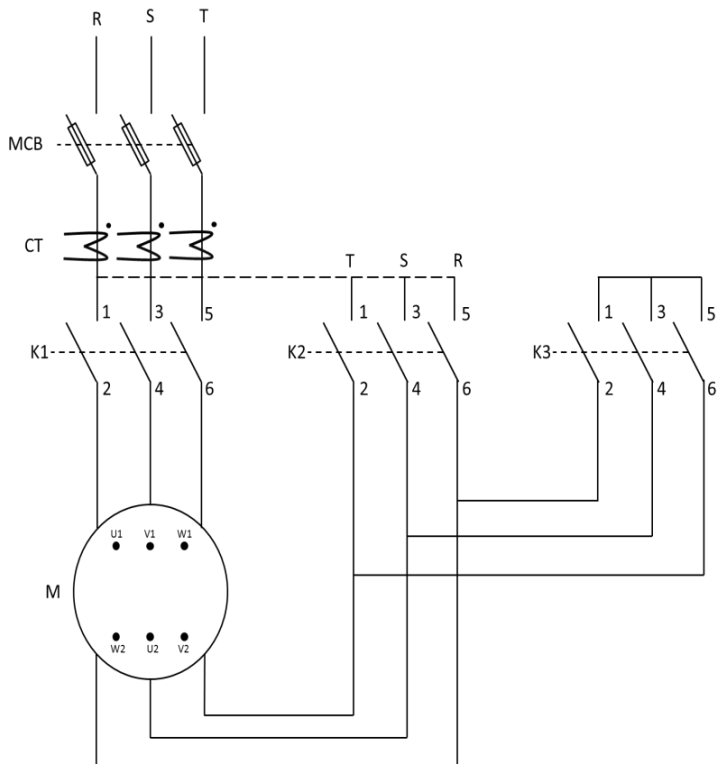
I. Tujuan :

1. Memahami prinsip kerja starting motor induksi 3 fasa dengan menggunakan metode *Wye-Delta*.
2. Memahami karakteristik arus *starting* metode *Wye-Delta*.
3. Mengenal rangkaian daya dan rangkaian kontrol dari metode *Wye-Delta*.

II. Peralatan yang dibutuhkan:

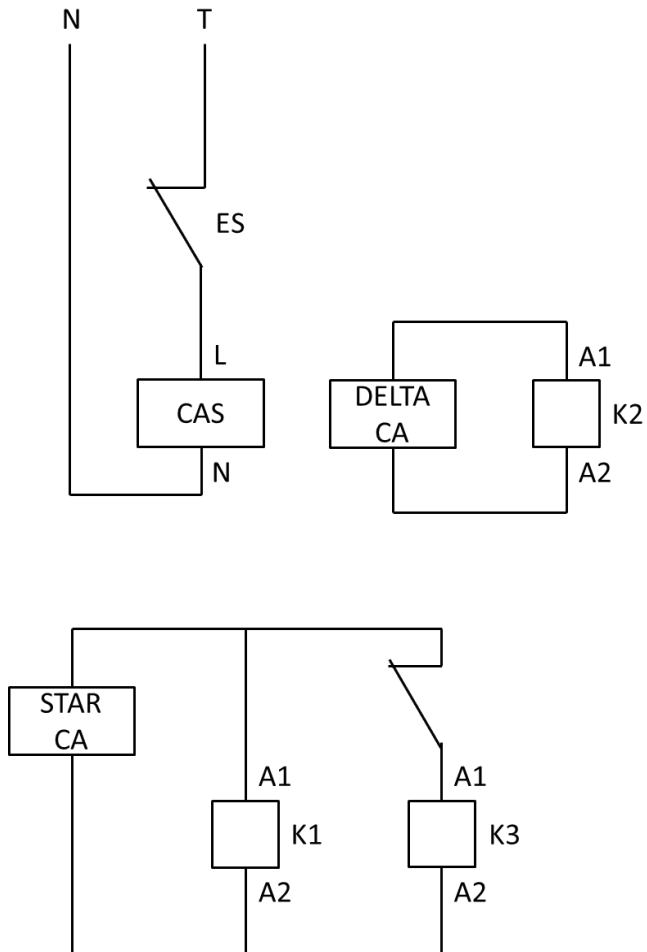
1. Peralatan keselamatan
2. Modul percobaan
3. Kabel jumper
4. Motor induksi 3 fasa
5. Modul '*Starting* Motor Induksi 3 Fasa'

III. Gambar rangkaian :
DAYA



Gambar 4.3 Rangkaian Daya Metode *Wye-Delta*

KONTROL



Gambar 4.4 Rangkaian Kontrol Metode *Wye-Delta*

KETERANGAN

MCB	: Miniature Circuit Breaker
ES	: Emergency Switch
M	: Motor
CAS	: Coil Activator Supply
STAR CA	: Star Coil Activator
DELTA CA	: DOL Coil Activator
CT	: Current Transformer

IV. Langkah percobaan:

1. Patuhi standar keamanan yang telah ditetapkan pada praktikum.
2. Pastikan sumber tegangan dalam keadaan mati.
3. Rangkailah rangkaian daya Wye-Delta seperti gambar 4.3.
4. Rangkailah rangkaian kontrol Wye-Delta seperti pada gambar 4.4.
5. Laporkan pada asisten terlebih dahulu untuk melakukan pengecekan.
6. Pastikan *Emergency Switch* dalam keadaan NC (*Normal Close*).
7. Nyalakan MCB.
8. Tekan tombol 'MENU' untuk masuk ke pilihan metode yang akan digunakan.
9. Tekan tombol 'DOWN' untuk mengarahkan cursor ke metode Star Delta, lalu tekan tombol 'MENU' untuk memilih metode.
10. Jalankan metode dengan menekan tombol 'MENU'
11. Amati lalu catat angka arus yang ditampilkan oleh LCD.
12. Tekan tombol 'MENU' untuk menghentikan motor.
13. Matikan MCB.
14. Analisa hasil pengukuran arus.

PERCOBAAN 3
STARTING MOTOR INDUKSI 3 FASA DENGAN
MENGGUNAKAN METODE *SOFT STARTING*

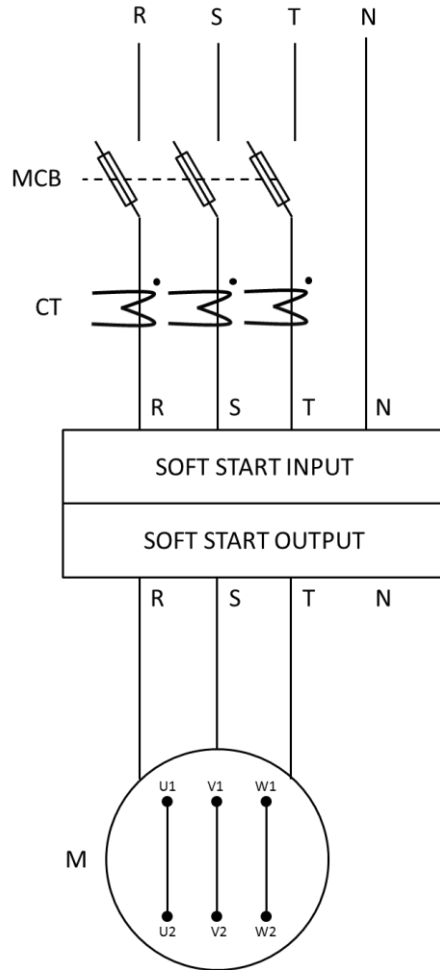
I. Tujuan :

1. Memahami prinsip kerja starting motor induksi 3 fasa dengan menggunakan metode *Soft Starting*.
2. Memahami karakteristik arus *starting* metode *Soft Starting*.
3. Mengenal rangkaian daya dan kontrol dari metode *Soft Starting*.

II. Peralatan yang dibutuhkan:

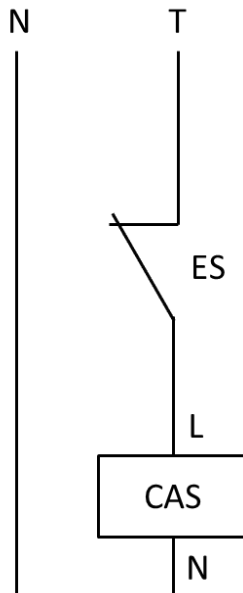
1. Peralatan keselamatan
2. Modul percobaan
3. Kabel jumper
4. Motor induksi 3 fasa
5. Modul '*Starting* Motor Induksi 3 Fasa'

**III. Gambar rangkaian :
DAYA**



Gambar 4.5 Rangkaian Daya Metode *Soft Starting*

KONTROL



Gambar 4.6 Rangkaian Kontrol Metode *Soft Starting*

KETERANGAN

MCB	: Miniature Circuit Breaker
ES	: Emergency Switch
CT	: Current Transformer
M	: Motor
CAS	: Coil Activator Supply

IV. Langkah percobaan:

1. Patuhi standar keamanan yang telah ditetapkan pada praktikum.
2. Pastikan sumber tegangan dalam keadaan mati.
3. Rangkailah rangkaian daya *Soft Starting* seperti gambar 4.5.

4. Rangkailah rangkaian kontrol *Soft Starting* seperti pada gambar 4.6.
5. Laporkan pada asisten terlebih dahulu untuk melakukan pengecekan.
6. Pastikan *Emergency Switch* dalam keadaan NC (*Normal Close*).
7. Nyalakan MCB.
8. Tekan tombol 'MENU' untuk masuk ke pilihan metode yang akan digunakan.
9. Tekan tombol 'DOWN' untuk mengarahkan cursor ke "Set Soft Start", lalu tekan tombol 'MENU' untuk memilih opsi, lalu atur waktu start untuk starting dengan menggunakan metode *Soft Starting* lalu tekan 'MENU'.
10. Arahkan cursor ke opsi 'Soft Starting' lalu tekan tombol 'MENU' untuk memilih metode, setelah itu tekan tombol 'MENU' sekali lagi untuk menjalankan motor.
11. Amati dan catat pergerakan arus pada LCD mulai 0% sampai 100% (*steady state*).
12. Tekan tombol 'MENU' untuk menghentikan motor.
13. Matikan MCB.
14. Analisa hasil pengukuran arus.

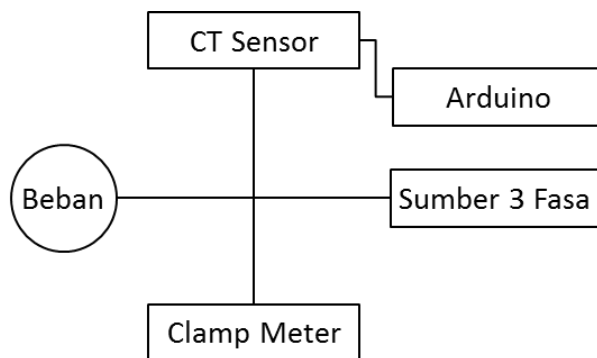
4.2. Pengujian Sistem

Pengujian merupakan salah satu langkah yang harus dilakukan untuk mengetahui apakah sistem yang telah dibuat sesuai dengan yang direncanakan. Kesesuaian sistem dapat dilihat dari keberhasilan hasil pengujian sistem yang telah dilakukan. Pengujian juga berfungsi untuk mengetahui kelebihan dan kekurangan sistem yang telah dibuat. Pengujian sistem tersebut meliputi :

1. Pengujian sesor arus CT-103
2. Pengujian rangkaian kontrol dari metode DOL (*Direct On-Line*), *Wye-Delta*, dan *Soft Starting*.
3. Pengujian motor induksi 3 fasa
4. Pengujian keseluruhan

4.2.1. Pengujian Sensor Arus CT-103

Pada pengujian sensor arus ini kami melakukan kalibrasi Sensor CT untuk mendapatkan tingkat akurasi pembacaan arus sebaik mungkin. Pengujian dilakukan kepada 3 sensor arus yang kami gunakan yang nantinya akan mewakili fasa R, S, T dengan menggunakan variasi arus gangguan yang berubah ubah nilainya. Pengujian ini dilakukan agar kita bisa mendapatkan nilai analog yang dibaca oleh arduino dari nilai tegangan *output* Sensor CT yang sudah berupa tegangan. Kalibrasi yang dilakukan ini dilakukan satu per satu untuk masing-masing Sensor CT. Beban yang kami gunakan untuk kalibrasi Sensor CT yaitu lampu yang dipasang pada setiap fasa nya. Skema pengujian Sensor CT dapat dilihat pada Gambar 4.7.



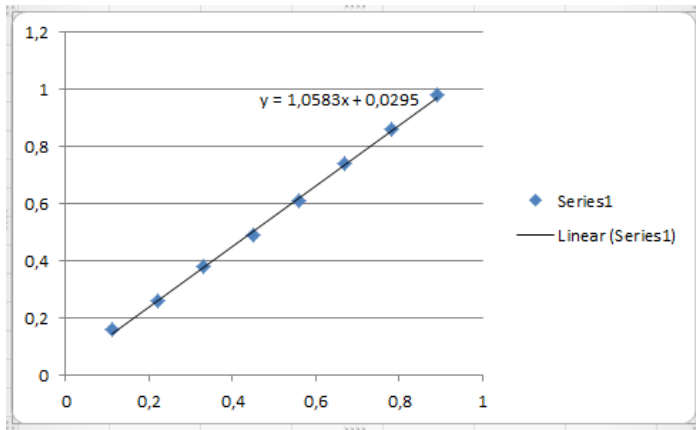
Gambar 4.7 Skema Pengujian Sensor CT.

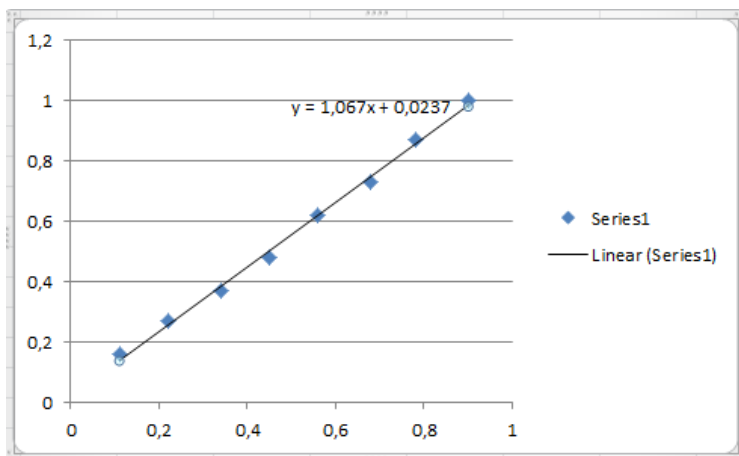
Dengan menggunakan skema pengujian Sensor CT seperti pada Gambar 4.7, maka akan didapatkan hasil pengukuran awal antara alat ukur *Clamp Meter* dan *Serial* pada arduino UNO yang terbebani lampu 25w hingga 200w ditampilkan seperti pada Tabel 4.1.

Tabel 4.1 Hasil Awal Pengujian Sensor CT

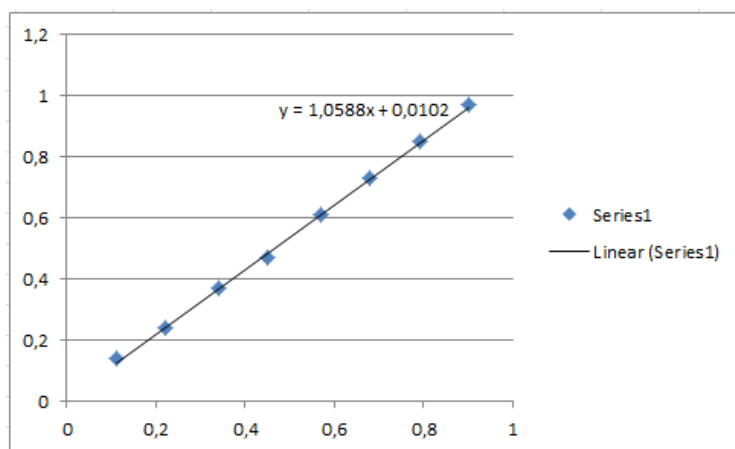
No.	Beban	CT 1		CT 2		CT 3	
		Clamp	Serial	Clamp	Serial	Clamp	Serial
1	25 w	0.11	0.16	0.11	0.16	0.11	0.14
2	50 w	0.22	0.26	0.22	0.27	0.22	0.24
3	75 w	0.33	0.38	0.34	0.37	0.34	0.37
4	100 w	0.45	0.49	0.45	0.48	0.45	0.47
5	125 w	0.56	0.61	0.56	0.62	0.57	0.61
6	150 w	0.67	0.74	0.68	0.73	0.68	0.73
7	175 w	0.78	0.86	0.78	0.87	0.79	0.85
8	200 w	0.89	0.98	0.90	1.00	0.90	0.97

Dari data pengujian CT sensor yang masuk ke Arduino dapat dibuat suatu persamaan karakteristik linierisasi sensor arus yang akan digunakan pada pembacaan nilai arus pada Arduino yang sesuai dengan nilai arus pengukuran menggunakan rumus *scatter* pada *Microsoft Excel* yang ditunjukkan pada Gambar 4.8 sampai Gambar 4.10.

**Gambar 4.8** Linierisasi Sensor CT 1 Fasa R Antara Clamp (sumbu Y) dengan Serial (sumbu X)



Gambar 4.9 Linierisasi Sensor CT 2 Fasa S Antara Clamp (sumbu Y) dengan Serial (sumbu X)

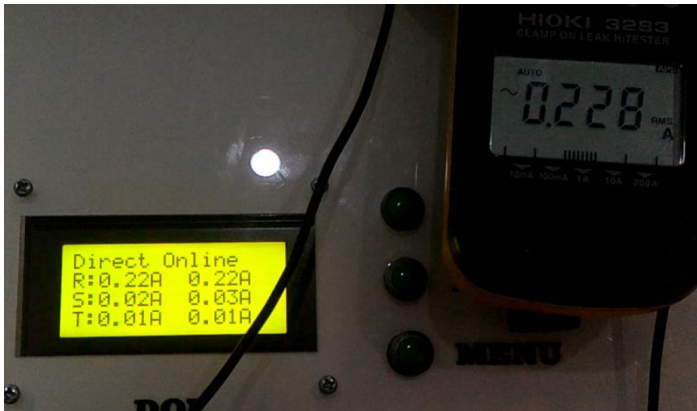


Gambar 4.10 Linierisasi Sensor CT 3 Fasa T Antara Clamp (sumbu Y) dengan Serial (sumbu X)

Setelah didapatkan persamaan linearnya dari masing-masing fasa kemudian dilakukan pengujian dengan menggunakan persamaan yang dimasukkan ke dalam program Arduino. Hasil pengujian dengan persamaan tersebut dilihat dengan mengukur arus hubung singkat yang

berbeda nilainya. Perhitungan presentase kesalahan pengukuran sensor arus didapat dari Persamaan 4.1. Untuk hasil pengujian presentase *error* sensor arus ini secara lengkap tiap fasa R, S dan T dapat dilihat pada Tabel 4.2. Dan gambar sampel pengambilan data akhir untuk pengujian Sensor CT dapat dilihat pada Gambar 4.11.

$$Error = \frac{Hasil\ alat\ ukur - Tampilan\ LCD}{Hasil\ alat\ ukur} \times 100\% \quad (4.1)$$



Gambar 4.11 Sampel Hasil Pengujian Sensor CT 1 (Fasa R) dengan Beban 25 watt

Tabel 4.2 Perbandingan Hasil Akhir dan Awal Serial Arduino UNO pada Pengujian Sensor CT

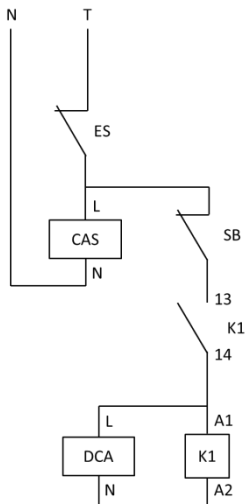
No.	Beban	CT 1		CT 2		CT 3	
		Akhir	Awal	Akhir	Awal	Akhir	Awal
1	25 w	0.12	0.16	0.12	0.16	0.10	0.14
2	50 w	0.22	0.26	0.23	0.27	0.20	0.24
3	75 w	0.34	0.38	0.33	0.37	0.33	0.37
4	100 w	0.45	0.49	0.44	0.48	0.43	0.47
5	125 w	0.57	0.61	0.58	0.62	0.57	0.61
6	150 w	0.70	0.74	0.69	0.73	0.69	0.73
7	175 w	0.82	0.86	0.83	0.87	0.81	0.85
8	200 w	0.94	0.98	0.96	1.00	0.93	0.97

Dari data Tabel 4.2 yang didapat dari pengujian, dapat dihitung rata-rata *error* pada sensor arus fasa R sebesar 3.6%, sensor arus fasa S sebesar 4.6%, dan sensor arus fasa T sebesar 4.1%. Error tersebut dihitung menggunakan persamaan 4.1 dengan data alat ukur *Clamp* pada tabel 4.1 dan data *Serial* akhir (Tampilan LCD) pada tabel 4.2. Perbedaan dari keluaran sensor tersebut dapat dikarenakan faktor internal maupun faktor eksternal. Faktor internal berasal dari karakteristik tiap komponen yang tidak semuanya memiliki standar yang baik. Faktor eksternal berasal dari tegangan sumber yang masuk tiap-tiap sensor tidak sama dan juga faktor dari kesalahan pembacaan alat ukur.

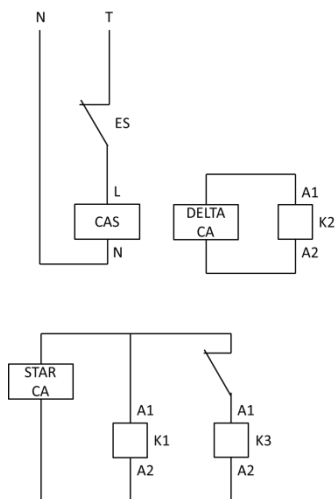
Sensor Arus CT-103 juga diprogram oleh mikrokontroler Arduino UNO untuk merecord nilai lonjakan arus dengan mengambil nilai *peak to peak* yang terjadi dan akan ditampilkan pada LCD.

4.2.2. Pengujian rangkaian kontrol dari metode DOL (*Direct On-Line*), *Wye-Delta*, dan *Soft Starting*.

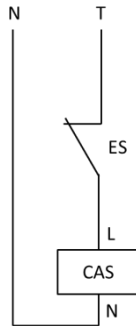
Rangkaian kontrol digunakan untuk mengendalikan alat sesuai yang kita harapkan. Pengujian rangkaian kontrol bertujuan untuk memastikan bahwa fungsi kontrol pada rangkaian berfungsi sebagaimana mestinya. Beberapa komponen yg diperhatikan pada rangkaian kontrol ini yaitu Kontaktor, *Coil Activator Supply*, MCB, dan *Emergency Switch* Pengujian rangkaian kontrol ini terdiri dari tiga metode yaitu DOL (*Direct On-Line*), *Wye-Delta*, dan *Soft Starting* yang bisa kita lihat pada gambar 4.12, gambar 4.13, dan gambar 4.14.



Gambar 4.12 Rangkaian Kontrol Metode DOL (*Direct On-Line*)

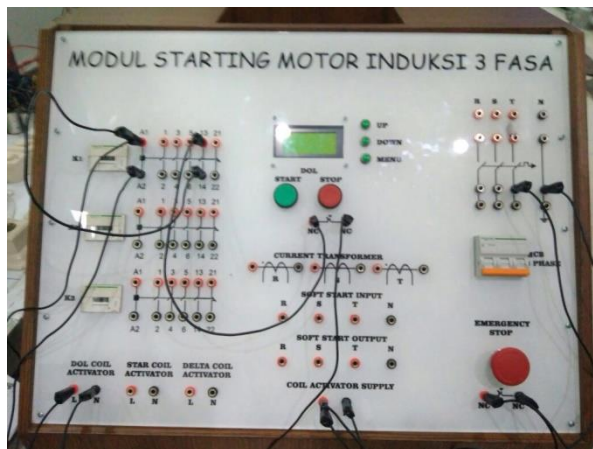


Gambar 4.13 Rangkaian Kontrol Metode *Wye-Delta*.

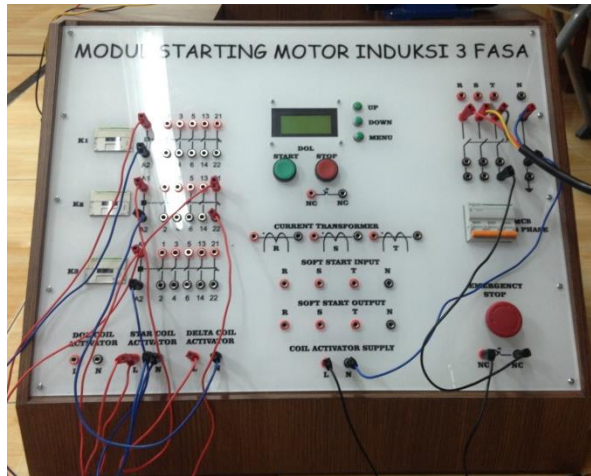


Gambar 4.14 Rangkaian Kontrol Metode *Soft Starting*.

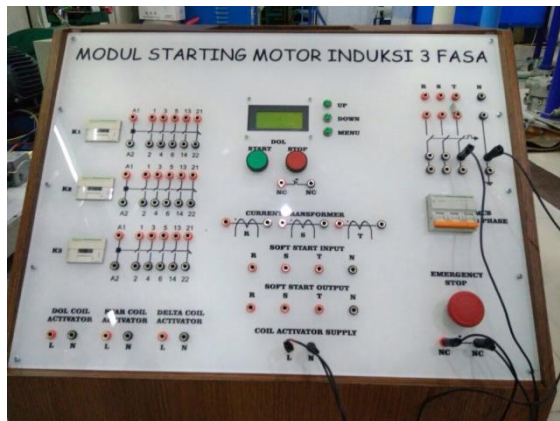
Hasil dari pengujian rangkaian kontrol ini adalah kontaktor dapat merespon dengan normal, dan mikrokontroler mendapatkan *supply* tegangan untuk menampilkan data pada LCD yang dapat kita lihat pada Gambar 4.15, Gambar 4.16, Gambar 4.17, dan Gambar 4.18.



Gambar 4.15 Rangkaian Kontrol pada Modul dengan Metode DOL.



Gambar 4.16 Rangkaian Kontrol pada Modul dengan Metode Wye-Delta



Gambar 4.17 Rangkaian Kontrol pada Modul dengan Metode Soft Starting



Gambar 4.18 Tampilan LCD setelah mendapat *Supply* Tegangan

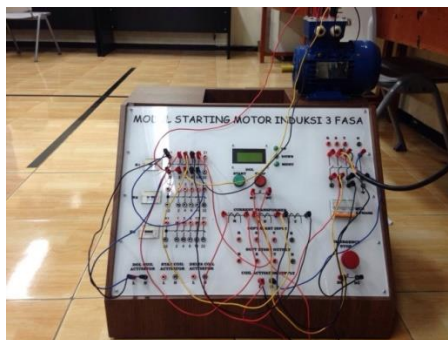
4.2.3. Pengujian Motor Induksi 3 fasa

Pengujian motor bertujuan untuk mengetahui bahwa motor dapat berputar dengan normal dengan suplai tegangan 380v metode DOL (*Direct On-Line*) secara *Wye* dan *Delta*. Dan hasil status motor dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil pengujian motor induksi 3 fasa

METODE	STATUS MOTOR
<i>Wye</i>	Berputar
<i>Delta</i>	Berputar

Rangkaian keseluruhan dari metode DOL pada modul Starting Motor Induksi 3 Fasa serta hasil pengukuran arus di setiap fasanya dapat kita lihat pada Gambar 4.19 dan Gambar 4.20.



Gambar 4.19 Pengujian Motor menggunakan Metode DOL

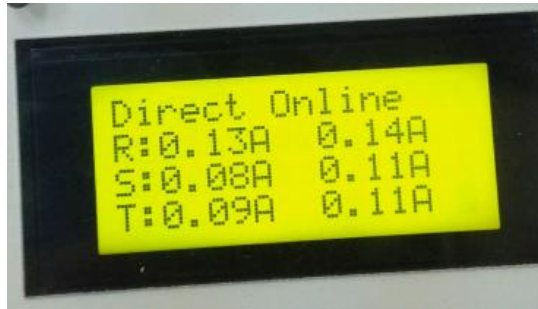


Gambar 4.20 Hasil pengukuran Arus fasa R, S, dan T pada Motor dengan metode DOL

4.2.4. Pengujian keseluruhan

Uji keluruhan adalah uji coba alat kami secara menyeluruh dan mengikuti prosedur percobaan yang ada. Uji keseluruhan dilakukan untuk mengetahui apakah alat yang kami buat bisa berjalan sesuai harapan. Pengujian ini menggunakan rangkaian daya yang dikombinasikan dengan rangkaian kontrol dari metode *starting* motor induksi 3 fasa secara DOL (*Direct On-Line*), *Wye-Delta*, dan *Soft Starting*. Hasil dari pengujian keseluruhan ini ada berupa nilai lonjakan arus pada masing-masing metode *starting*, yang dimana nilai lonjakan arus tersebut akan dibandingkan dengan masing-masing nilai arus pada setiap metode.

Dari gambar 4.21 dan gambar 4.22 dapat kita lihat perbandingan lonjakan arus *starting* dari metode DOL secara *Wye* dan DOL secara *Delta*. Dari gambar 4.23 dan gambar 4.24, dapat kita lihat perubahan nilai lonjakan arus *starting* dari metode *Wye-Delta*. Pada tabel 4.4, bisa kita lihat tahap kenaikan arus pada metode *Soft Starting*.



Gambar 4.21 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode DOL (*Direct On-Line*) secara Wye.



Gambar 4.22 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode DOL (*Direct On-Line*) secara Delta



Gambar 4.23 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode Wye-Delta saat Wye.



Gambar 4.24 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode *Wye-Delta* saat *Delta*.

Tabel 4.4 Hasil pengukuran arus oleh mikrokontroler Arduino UNO dengan metode *Soft Starting*

detik	Arus R (A)	Arus S (A)	Arus T (A)
1	0.05	0.11	0.05
2	0.22	0.41	0.27
3	0.26	0.46	0.29
4	0.27	0.47	0.33
5	0.36	0.46	0.37
6	0.35	0.46	0.39
7	0.40	0.48	0.49
8	0.31	0.34	0.36
9	0.48	0.40	0.45
10	0.51	0.40	0.42
11	0.47	0.41	0.42
12	0.50	0.42	0.42
13	0.48	0.39	0.43
14	0.49	0.39	0.43
15	0.35	0.31	0.55

BAB V

PENUTUP

Bab penutup berisi tentang kesimpulan yang didapat selama proses pengujian alat “Starting Motor Induksi 3 Fasa” serta saran untuk pengembangan alat ini kedepannya.

5.1 Kesimpulan

Kesimpulan yang bisa saya ambil dari buku ini adalah starting motor induksi 3 fasa dapat dibedakan menjadi 3 metode, antara lain:

1. DOL (*Direct On-Line*)
2. *Wye-Delta*
3. *Soft Starting*

Pada setiap metode mempunyai prosedur untuk menjalankan Modul Starting Motor Induksi 3 Fasa menggunakan modul praktikum. Sebelum mengikuti prosedur dalam modul praktikum, sensor arus CT akan diuji terlebih dahulu melalui kalibrasi agar perhitungan menjadi akurat.

Pada pengujian sensor arus ini kami melakukan kalibrasi Sensor CT untuk mendapatkan tingkat akurasi pembacaan arus sebaik mungkin. Dari kalibrasi tersebut akan didapatkan hasil awal berupa nilai pembacaan arus yang belum akurat.

Setelah mendapatkan hasil awal, maka akan dibuat persamaan karakteristik linierisasi sensor arus CT. Setelah itu akan diproses melalui Arduino UNO. Setelah diproses pada Arduino Uno, maka akan ditemukan hasil akhir kalibrasi disetiap fasanya pada sensor CT. Setelah pengujian sensor CT, dapat dihitung rata-rata *error* pada sensor CT untuk sensor arus fasa R sebesar 3.6%, sensor arus fasa S sebesar 4.6%, dan sensor arus fasa T sebesar 4.1%.

Setelah kalibrasi telah sukses, selanjutnya kami melakukan percobaan pada modul praktikum *Starting* Motor Induksi 3 Fasa dengan ketiga metode tersebut. Hasil percobaan tersebut adalah arus lonjakan yang terjadi pada setiap metode *starting*.

Hasil dari perbandingan arus lonjakan yang terjadi pada setiap metode *starting* motor menurut analisa saya yaitu sebagai berikut.

1. **DOL (*Direct On-Line*)** : Motor hanya mendapatkan satu kali *supply* tegangan kejut sebesar 380v, tetapi arus pada saat *steady*

state tidak terlalu turun dari arus lonjakannya. Sehingga motor terkejut karena mendapatkan *supply* tegangan yang besar pada saat *starting* sehingga arusnya kecil dan putaran motor tidak maksimal. Hal ini justru membahayakan motor dan menyebabkan motor cepat untuk mengalami kerusakan.

2. **Wye-Delta** : Motor mendapatkan dua kali *supply* tegangan kejut dengan tegangan kejut pertama yang besar (380v) sehingga arusnya kecil agar putaran motor pelan. Setelah itu dalam *delay* tertentu motor akan mendapatkan *supply* tegangan kejut kedua yang lebih kecil dari *supply* tegangan yang pertama yaitu sebesar 220v untuk mendapatkan arus yang lebih besar dari *supply* tegangan pertama, maka motor akan berputar semakin cepat sehingga memiliki kecepatan yang maksimal. Metode ini lebih awet digunakan dalam industri dikarenakan motor akan lebih awet dan tidak cepat rusak.
3. **Soft Starting** : Metode ini yang sangat aman untuk digunakan dikarenakan adanya tahapan *supply* tegangan ke motor, sehingga lonjakan arus akan menurun derastis dan putaran motor akan menjadi lebih halus saat kondisi berhenti hingga kondisi *steady state*.

5.2 Saran

Saran dari saya adalah sebaiknya alat ini agar disempurnakan lagi dalam pembacaan arus agar prosentase *error* dalam pembacaannya menjadi lebih kecil sehingga pembacaan arus menjadi lebih akurat.

Sebaiknya untuk melakukan *rewinding* pada motor induksi 3 fasa jika memakai motor induksi 3 fasa berkapasitas dibawah 3 HP untuk mendapatkan *starting* dengan metode *Wye-Delta* yang lebih sempurna. Dan perbaiki *rewinding* motor agar arus lonjakan lebih terlihat jelas perbedaannya.

Saran saya yang terakhir adalah tetap berjuang dan jangan lupa berdoa kepada Allah SWT.

DAFTAR PUSTAKA

- [1] Belajar Elektronika.(2016, 11 Januari). Pengertian Motor Listrik 3 Fasa dan Prinsip Kerjanya. Diperoleh 22 Mei 2017, <http://belajarelektronika.net/pengertian-motor-listrik-3-fasa/>
- [2] Almon.(2008, 13 Desember). Metode Starting Motor Induksi. Diperoleh 22 Mei 2017, <http://almon-r.blogspot.co.id/2008/12/metode-starting-motor-induksi.html>
- [3] Motor Induksi 3 Fasa.(2016, 23 April). Metode Pengasutan Motor Induksi 3 Fasa. Diperoleh 22 Mei 2017, <http://motorinduksi3phasa.blogspot.co.id/2016/04/metode-pengasutan-motor-induksi-3-phasa.html>
- [4] Daryanto.2016.*Konsep Dasar Teknik Elektronika Kelistrikan*.Bandung: Alfabeta.
- [5] Arindya, Radita.2013.*Penggunaan dan Pengaturan Motor Listrik*. Yogyakarta: Graha Ilmu.

-----*Halaman ini sengaja dikosongkan*-----

LAMPIRAN A

A.1 Listing Program Pada Arduino

```
//#include <EmonLib.h>
//EnergyMonitor arus;

#include <LiquidCrystal.h>
LiquidCrystal lcd (A5, A4, A3, 13, 12, 2);

////definisi ADC

// Voltage Reference: AVCC pin
#define ADC_VREF_TYPE ((0<<REFS1) | (1<<REFS0) | (0<<ADLAR))

// Voltage Reference: AREF pin
#define ADC_VREF_TYPE ((0<<REFS1) | (0<<REFS0) | (0<<ADLAR))

// Voltage Reference: Int., cap. on AREF
#define ADC_VREF_TYPE ((1<<REFS1) | (1<<REFS0) | (0<<ADLAR))

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX = adc_input | ADC_VREF_TYPE;
    // Delay needed for the stabilization of the ADC input voltage
    //delay_us(10);
    //delayMicroseconds(10);
    // Start the AD conversion
    ADCSRA |= (1 << ADSC);
    // Wait for the AD conversion to complete
    while ((ADCSRA & (1 << ADIF)) == 0);
    ADCSRA |= (1 << ADIF);
    return ADCW;
}
void adcInit(void) {
    // ADC initialization
```

```

// ADC Clock frequency: 1000,000 kHz
// ADC Voltage Reference: AVCC pin
// ADC Auto Trigger Source: ADC Stopped
// Digital input buffers on ADC0: On, ADC1: On, ADC2: On, ADC3:
On
// ADC4: On, ADC5: On, ADC6: On, ADC7: On

//DIDR0 = (0 << ADC7D) | (0 << ADC6D) | (0 << ADC5D) | (0 <<
ADC4D) | (0 << ADC3D) | (0 << ADC2D) | (0 << ADC1D) | (0 <<
ADC0D);

DIDR0 = (0 << ADC5D) | (0 << ADC4D) | (0 << ADC3D) | (0 <<
ADC2D) | (0 << ADC1D) | (0 << ADC0D);
ADMUX = ADC_VREF_TYPE;
ADCSRA = (1 << ADEN) | (0 << ADSC) | (0 << ADIF) | (0 << ADIE) | (1 << ADPS2) | (0 << ADPS1) | (0 << ADPS0);
ADCSRB = (0 << ADTS2) | (0 << ADTS1) | (0 << ADTS0);

}

////

//output pin
const int StartSTARpin = 5;
const int StartDELTApin = 4;

const int StartRSSpin = 11;
const int StartSSSpin = 10;
const int StartTSSpin = 9;

//input pin
const int UPpin = 3;
const int DOWNpin = 1;
const int MENUpin = 0;

//analog pin
/*
const int CurrentRpin = A0;
const int CurrentSpin = A1;

```

```

const int CurrentTpin = A2;
*/
#define currentRpin 0
#define currentSpin 1
#define currentTpin 2

const int ZCRpin = 6;
const int ZCSpin = 7;
const int ZCTpin = 8;

String txtDOL    = "Direct Online ";
String txtStartDelta = "Star Delta ";
String txtSoftStart = "Soft Starting ";
String txtSet     = "Set Soft Start";

#include <FlexiTimer2.h>

int dutySet = 10;

int cDutyT = 0;
int dutyT = dutySet;
boolean gasspollT = false;

int cDutyS = 0;
int dutyS = dutySet;
boolean gasspollS = false;

int cDutyR = 0;
int dutyR = dutySet;
boolean gasspollR = false;

void onDuty(void) {
  if (gasspollT == true) {
    cDutyT++;
    if (cDutyT > dutyT) {
      digitalWrite(StartTSSpin, LOW);
    }
  }
  else {

```

```

        digitalWrite(StartTSSpin, HIGH);
    }
    if (cDutyT > 10)cDutyT = 0;
}

if (gasspollS == true) {
    cDutyS++;
    if (cDutyS > dutyS) {
        digitalWrite(StartSSSpin, LOW);
    }
    else {
        digitalWrite(StartSSSpin, HIGH);
    }
    if (cDutyS > 10)cDutyS = 0;
}

if (gasspollR == true) {
    cDutyR++;
    if (cDutyR > dutyR) {
        digitalWrite(StartRSSpin, LOW);
    }
    else {
        digitalWrite(StartRSSpin, HIGH);
    }
    if (cDutyR > 10)cDutyR = 0;
}
}

void setup()
{
    //Setting I/O
    pinMode(UPpin, INPUT_PULLUP);
    pinMode(DOWNpin, INPUT_PULLUP);
    pinMode(MENUpin, INPUT_PULLUP);

    pinMode(StartSTARpin, OUTPUT);
    pinMode(StartDELTApin, OUTPUT);
    pinMode(StartRSSpin, OUTPUT);
    pinMode(StartSSSpin, OUTPUT);
}

```



```

pinMode(StartTSSpin, OUTPUT);

digitalWrite(StartSTARpin, HIGH);
digitalWrite(StartDELTApin, HIGH);
digitalWrite(StartRSSpin, HIGH);
digitalWrite(StartSSSpin, HIGH);
digitalWrite(StartTSSpin, HIGH);

Serial.begin(115200);
Serial.println("Programmable Motor Starting");
Serial.println("Activated !!");
Serial.end();

//Timer1.initialize(50000);
//FlexiTimer2::set(0.5, 1.0 / 2000, onDuty); // call every 500 1ms
"ticks"
FlexiTimer2::set(1, onDuty); // MsTimer2 style is also supported

//FlexiTimer2::start();

/* add setup code here */
adcInit();

//arus.current(CurrentRpin, 111.1);

lcd.begin(16, 4);
lcd.clear();
/*
lcd.setCursor(0, 0);
lcd.print("Ya..Meitahh..");
lcd.setCursor(0, 1);
lcd.print("test");
lcd.setCursor(0, 2);
lcd.print("1");
lcd.setCursor(0, 3);
lcd.print("oke");
*/
}

```

```

int cTombol = 0;
long tombolMillis = 0;
long dMillis = 0;
signed char baris = 0;
boolean sBlink = false;

///punyae soft
long softMillisR = 0;
long softMillisS = 0;
long softMillisT = 0;

float lamaStart = 25.0;

boolean softEnable = false;

boolean atHome = true;

/////untuk debounce
int dMax=5000, dLimit=4500, dValue=4000;

/////Arus
float arusR = 0.0, arusS = 0.0, arusT = 0.0;
float arusMaxR = 0.0, arusMaxS = 0.0, arusMaxT = 0.0;
int maxR = 0, maxS = 0, maxT = 0;
int minR = 1023, minS = 1023, minT = 1023;

void loop()
{
  if (millis() - dMillis > 1000) {
    dMillis = millis();
    if (atHome) {
      displayHome();
    }
    else {
      tampilan();
    }
  }
}

if (atHome == true) {

```

```

if (digitalRead(MENUpin) == LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
        while (digitalRead(MENUpin) == LOW) {
            lcd.clear();
        }
        atHome = false;
        tampilan();
    }
}
else {
    cTombol = 0;
}
}
else {
    readKey();
}

}

void displayValue(void) {
    if (millis() - dMillis > 500) {
        dMillis = millis();

        lcd.setCursor(0, 1);
        lcd.print("R:");
        lcd.print(arusR);
        lcd.print("A ");
        lcd.print(arusMaxR);
        lcd.print("A ");

        lcd.setCursor(0, 2);
        lcd.print("S:");
        lcd.print(arusS);
        lcd.print("A ");
        lcd.print(arusMaxS);
        lcd.print("A ");
    }
}

```

```

    lcd.setCursor(0, 3);
    lcd.print("T:");
    lcd.print(arusT);
    lcd.print("A ");
    lcd.print(arusMaxT);
    lcd.print("A ");
}
}

void modeDOL(void) {

    while (digitalRead(MENUPin) == LOW) {
        lcd.clear();
    }
    lcd.setCursor(0, 0);
    lcd.print(txtDOL);
    while (1) {
        readArus();
        displayValue();
        //double Irms = arus.calcIrms(1480);
        //lcd.setCursor(0, 1);
        //lcd.print("Arus R:");
        //lcd.print(Irms * 230.0);

        if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
            cTombol++;
            if (cTombol >= dMax)cTombol = dLimit;
            if (cTombol == dValue) {
                //atHome = true;
                lcd.clear();
                tampilan();
                arusMaxR = 0.0;
                arusMaxS = 0.0;
                arusMaxT = 0.0;

                while (digitalRead(UPpin) == LOW && digitalRead(DOWNpin)
== LOW);

```

```

        break;
    }
}
else {
    cTombol = 0;
}
}
}
}

```

```

boolean statusStarDelta = false;
uint8_t sdc = 0;
long starDeltaMillis = 0;
long cDownMillis = 0;
long timeStarDelta = 5000;//order mili detik
void modeStarDelta(void) {
    while (digitalRead(MENUpin) == LOW) {
        lcd.clear();
    }
    lcd.setCursor(0, 0);
    lcd.print(txtStartDelta);
    lcd.setCursor(13, 0);
    lcd.print("[");
    lcd.print(timeStarDelta / 1000);
    lcd.print("]");
}

```

```

while (1) {

    if (digitalRead(MENUpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {
            lcd.begin(16, 4);
            //lcd.clear();
            //lcd.setCursor(0, 0);
            //lcd.print("          ");
            if (statusStarDelta == false) {
                sdc = 0;
            }
        }
    }
}

```

```

        //lcd.begin(16, 4);
        lcd.setCursor(0, 0);
        lcd.print("STAR ");
        lcd.setCursor(13, 0);
        lcd.print('I');
        lcd.print(timeStarDelta / 1000);
        lcd.print('J');
        statusStarDelta = true;
        starDeltaMillis = millis();
        cDownMillis = millis();
    }
    else {
        statusStarDelta = false;
        //lcd.begin(16, 4);
        lcd.setCursor(0, 0);
        lcd.print(txtStartDelta);
        lcd.setCursor(13, 0);
        lcd.print('I');
        lcd.print(timeStarDelta / 1000);
        lcd.print('J');
    }
}
}
else if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
        sdc = 0;
        lcd.begin(16, 4);
        lcd.clear();
        digitalWrite(StartSTARpin, HIGH);
        digitalWrite(StartDELTApin, HIGH);
        statusStarDelta = false;
        while (digitalRead(UPpin) == LOW && digitalRead(DOWNpin)
== LOW);
        tampilan();
        arusMaxR = 0.0;
        arusMaxS = 0.0;
    }
}

```

```

        arusMaxT = 0.0;
        break;
    }
}
else {
    cTombol = 0;
}
readArus();
displayValue();

if (statusStarDelta == true) {

    digitalWrite(StartSTARpin, LOW);
    if ( millis() - starDeltaMillis >= timeStarDelta && sdc <= 5) {
        digitalWrite(StartDELTApin, LOW);
        lcd.setCursor(5, 0);
        lcd.print("DELTA ");
    }
    //multitask
    if (millis() - cDownMillis >= 1000) {
        cDownMillis = millis();
        sdc++;
        if (sdc >= 25)sdc = 20;
        lcd.setCursor(0, 0);
        lcd.print("STAR ");

        if (sdc < 5) {
            lcd.setCursor(13, 0);
            lcd.print('[');
            lcd.print((timeStarDelta / 1000) - sdc);
            lcd.print(']');
            lcd.setCursor(15, 0);
        }
        else {
            lcd.setCursor(13, 0);
            lcd.print("RUN");
        }
    }
}

```

```

//blink panah
if (millis() - tombolMillis > 333) {
    tombolMillis = millis();
    sBlink = !sBlink;
    if (sBlink == true) {
        lcd.setCursor(11, 0);
        lcd.print(">>");
    }
    else if (sBlink == false) {
        lcd.setCursor(11, 0);
        lcd.print(" ");
    }
}
}
else {
    sdc = 0;
    digitalWrite(StartSTARpin, HIGH);
    digitalWrite(StartDELTApin, HIGH);

    if (millis() - cDownMillis >= 1000) {
        cDownMillis = millis();

        lcd.setCursor(0, 0);
        lcd.print(txtStartDelta);
        //lcd.setCursor(15, 0);
        //lcd.print(timeStarDelta );
    }
}
//blink kotak
if (millis() - tombolMillis > 333) {
    tombolMillis = millis();
    sBlink = !sBlink;
    if (sBlink == true) {
        lcd.setCursor(11, 0);
        lcd.print("<>");
    }
    else if (sBlink == false) {
        lcd.setCursor(11, 0);
        lcd.print(" ");
    }
}

```



```

    }
  }
}
}
}

```

```

long progressMillis = 0;
int cProgress = 0;

```

```

void modeSoftStart(void) {
  while (digitalRead(MENUpin) == LOW) {
    lcd.clear();
  }
  cProgress = 0;
  lcd.setCursor(0, 0);
  lcd.print(txtSoftStart);

```

```

while (1) {
  readArus();
  displayValue();

```

```

  if (digitalRead(MENUpin) == LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
      lcd.begin(16, 4);
      lcd.setCursor(0, 0);
      lcd.print('[');
      lcd.print(cProgress);
      lcd.print("%]");

```

```

      if (softEnable == false) {
        softEnable = true;
        FlexiTimer2::start();
      }
      else if (softEnable == true) {

        softEnable = false;
        FlexiTimer2::stop();

```

```

    cProgress = 0;

    dutySet = 10;

    cDutyT = 0;
    dutyT = dutySet;
    gasspollT = false;

    cDutyS = 0;
    dutyS = dutySet;
    gasspollS = false;

    cDutyR = 0;
    dutyR = dutySet;
    gasspollR = false;

    digitalWrite(StartRSSpin, HIGH);
    digitalWrite(StartSSSpin, HIGH);
    digitalWrite(StartTSSpin, HIGH);

    lcd.setCursor(0, 0);
    lcd.print(txtSoftStart);
}
}
}
else if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {

        softEnable = false;
        FlexiTimer2::stop();

        dutySet = 10;

        cDutyT = 0;
        dutyT = dutySet;

```

```

gasspollT = false;

cDutyS = 0;
dutyS = dutySet;
gasspollS = false;

cDutyR = 0;
dutyR = dutySet;
gasspollR = false;

digitalWrite(StartRSSpin, HIGH);
digitalWrite(StartSSSpin, HIGH);
digitalWrite(StartTSSpin, HIGH);

digitalWrite(StartRSSpin, HIGH);
digitalWrite(StartSSSpin, HIGH);
digitalWrite(StartTSSpin, HIGH);

lcd.clear();
while (digitalRead(UPpin) == LOW && digitalRead(DOWNpin)
== LOW);
    tampilan();
    arusMaxR = 0.0;
    arusMaxS = 0.0;
    arusMaxT = 0.0;
    break;
}
}
else {
    cTombol = 0;
}

////////////////////////////////////

if (softEnable) {
    //Punya e T
    if (gasspollT == false) {
        if (digitalRead(ZCTpin) == LOW) {
            gasspollT = true;

```

```

    softMillisT = millis();
}
}
if (dutyT > 0) {
    if (millis() - softMillisT > (lamaStart * 100)) {
        softMillisT = millis();
        dutyT--;
    }
}
//Punya e S
if (gasspollS == false) {
    if (digitalRead(ZCSpin) == LOW) {
        gasspollS = true;
        softMillisS = millis();
    }
}
if (dutyS > 0) {
    if (millis() - softMillisS > (lamaStart * 100)) {
        softMillisS = millis();
        dutyS--;
    }
}
//Punya e R
if (gasspollR == false) {
    if (digitalRead(ZCRpin) == LOW) {
        gasspollR = true;
        softMillisR = millis();
    }
}
if (dutyR > 0) {
    if (millis() - softMillisR > (lamaStart * 100)) {
        softMillisR = millis();
        dutyR--;
    }
}
//progres bar
if (millis() - progressMillis > (lamaStart * 10)) {
    progressMillis = millis();
    cProgress++;
}

```

```

        if (cProgress >= 100)cProgress = 100;
        lcd.setCursor(0, 0);
        lcd.print('I');

        lcd.print(cProgress);
        lcd.print("%");

        lcd.setCursor(5 + (cProgress/10), 0);
        lcd.print('#');
    }
}
else {
    dutySet = 10;
    gasspollR = false;
    gasspollS = false;
    gasspollT = false;
    digitalWrite(StartRSSpin, HIGH);
    digitalWrite(StartSSSpin, HIGH);
    digitalWrite(StartTSSpin, HIGH);
}
}
}

void modeSetSoftStart(void) {
    float lamaNYA=0.0;

    lamaNYA = lamaStart;
    while (digitalRead(MENUpin) == LOW) {
        lcd.clear();
    }
    lcd.setCursor(0, 0);
    lcd.print(txtSet);
    lcd.setCursor(0, 1);
    lcd.print("Time: ");
    lcd.print(lamaNYA,1);
    lcd.print("second ");

    while (1) {

```

```

    if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {
            lcd.setCursor(0, 2);
            lcd.print("Use: ");
            lcd.print(lamaStart,1);
            lcd.print(" second ");
            delay(2500);
            lcd.clear();
            while (digitalRead(UPpin) == LOW && digitalRead(DOWNpin)
== LOW);
            break;
            tampilan();
        }
    }
    else if (digitalRead(MENUpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {
            lamaStart = lamaNYA;
            lcd.setCursor(0, 2);
            lcd.print("Set: ");
            lcd.print(lamaStart,1);
            lcd.print(" second ");
            delay(2500);
            lcd.clear();
            break;
            tampilan();
        }
    }
    else if (digitalRead(UPpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax/2)cTombol = dLimit/2;
        if (cTombol == dValue/2) {
            lamaNYA=lamaNYA + 0.1;

```

```

        if (lamaNYA >= 11.0)lamaNYA = 11.0;
        lcd.setCursor(0, 0);
        lcd.print(txtSet);
        lcd.setCursor(0, 1);
        lcd.print("Time: ");
        lcd.print(lamaNYA,1);
        lcd.print("second ");
    }
}
else if (digitalRead(DOWNpin) == LOW) {
    cTombol++;
    if (cTombol >= dMax/2)cTombol = dLimit/2;
    if (cTombol == dValue/2) {
        lamaNYA = lamaNYA - 0.1;
        if (lamaNYA <= 4.0)lamaNYA = 4.0;
        lcd.setCursor(0, 0);
        lcd.print(txtSet);
        lcd.setCursor(0, 1);
        lcd.print("Time: ");
        lcd.print(lamaNYA,1);
        lcd.print("second ");
    }
}
else {
    cTombol = 0;
}

}
}

void readKey(void) {
    if (digitalRead(MENUpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {

            switch (baris){
                case 0:
                    modeDOL();

```

```

        break;
    case 1:
        modeStarDelta();
        break;
    case 2:
        modeSoftStart();
        break;
    case 3:
        modeSetSoftStart();
        break;
    default:
        break;
}

}

}
else if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
        atHome = true;
        lcd.clear();
    }
}
else if (digitalRead(UPpin) == LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
        lcd.setCursor(0, baris);
        lcd.print(' ');

        baris--;
        if (baris < 0) baris = 3;
    }
}
else if (digitalRead(DOWNpin) == LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;

```



```

    if (cTombol == dValue) {
        lcd.setCursor(0, baris);
        lcd.print(' ');

        baris++;
        if (baris > 3) baris = 0;
    }
}
else {
    cTombol = 0;
}
///  

//blink kursor
if (millis() - tombolMillis > 333) {
    tombolMillis = millis();
    sBlink = !sBlink;
    if (sBlink == true) {
        lcd.setCursor(0, baris);
        lcd.print('>');
    }
    else if (sBlink == false) {
        lcd.setCursor(0, baris);
        lcd.print(' ');
    }
}
}
}

void displayHome(void) {
    lcd.setCursor(0, 0);
    lcd.print(" MODUL STARTING ");
    lcd.setCursor(0, 1);
    lcd.print("MOTOR IND 3 FASA");
    lcd.setCursor(0, 2);
    lcd.print(">> DANGEROUS! <<");
    lcd.setCursor(0, 3);
    lcd.print("--High Voltage--");

}

void tampilan(void)

```

```

{
  lcd.setCursor(1, 0);
  lcd.print(txtDOL);
  lcd.setCursor(1, 1);
  lcd.print(txtStartDelta);
  lcd.setCursor(1, 2);
  lcd.print(txtSoftStart);
  lcd.setCursor(1, 3);
  lcd.print(txtSet);
}

```

```

double akarDua = 1.414213562373095;
float faktorArus = 1.35;
long aMillis = 0;
int tempR=0, tempS=0, tempT=0;
int cSampling = 0;
int cArus = 0;

```

```

long rataMinR=0, rataMaxR=0, rataMinS = 0, rataMaxS = 0, rataMinT
= 0, rataMaxT = 0;

```

```

float nolAmpere = 0.06;

```

```

void readArus(void) {

  if (micros() - aMillis >= 1000) {
    aMillis = micros();
    //signed int AA = analogRead(CurrentRpin);
    //signed int BB = analogRead(CurrentSpin);
    //signed int CC = analogRead(CurrentTpin);
    int AA = read_adc(currentRpin);
    int BB = read_adc(currentSpin);
    int CC = read_adc(currentTpin);

    if (AA < minR)minR = AA;
    else if (AA > maxR)maxR = AA;

    if (BB < minS)minS = BB;

```

```

else if (BB > maxS)maxS = BB;

if (CC < minT)minT = CC;
else if (CC > maxT)maxT = CC;

cSampling++;

if (cSampling > 20) {
    cSampling = 0;

    rataMaxR += maxR;
    rataMinR += minR;
    maxR = 0;
    minR = 1023;

    rataMaxS += maxS;
    rataMinS += minS;
    maxS = 0;
    minS = 1023;

    rataMaxT += maxT;
    rataMinT += minT;
    maxT = 0;
    minT = 1023;

    cArus++;
}
if (cArus > 25) {
    cArus = 0;

    rataMaxR /= 25;
    rataMinR /= 25;
    float vppR = (rataMaxR - rataMinR) * 4.89 / 1023;
    rataMinR = 0;
    rataMaxR = 0;
    float vrmsR = vppR / 2 * akarDua;
    arusR = vrmsR-(nolAmpere + 0.03);
    arusR *= faktorArus;

```

```

rataMaxS /= 25;
rataMinS /= 25;
float vppS = (rataMaxS - rataMinS) * 4.89 / 1023;
rataMinR = 0;
rataMaxR = 0;
float vrmsS = vppS / 2 * akarDua;
arusS = vrmsS - (nolAmpere + 0.03);
arusS *= faktorArus;

rataMaxT /= 25;
rataMinT /= 25;
float vppT = (rataMaxT - rataMinT) * 4.89 / 1023;
rataMinT = 0;
rataMaxT = 0;
float vrmsT = vppT / 2 * akarDua;
arusT = vrmsT - nolAmpere;
arusT *= faktorArus;

if (arusR < 0) arusR *= -1;
if (arusS < 0) arusS *= -1;
if (arusT < 0) arusT *= -1;

if (arusR > arusMaxR) arusMaxR = arusR;
if (arusS > arusMaxS) arusMaxS = arusS;
if (arusT > arusMaxT) arusMaxT = arusT;

}

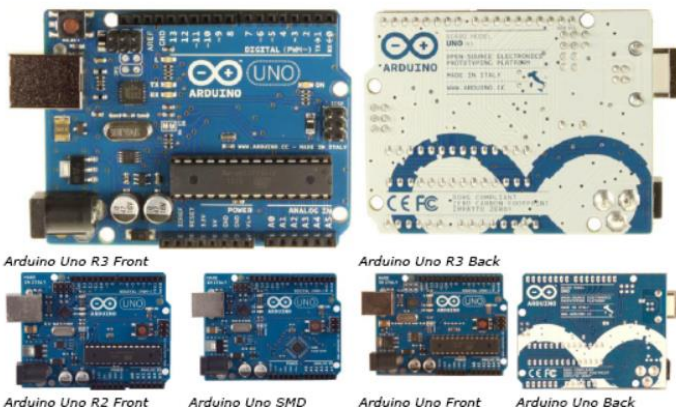
}

```

LAMPIRAN B

B.1 Datasheet Arduino Uno

Arduino Uno



Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

[Revision 2](#) of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

[Revision 3](#) of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V

Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Schematic & Reference Design

EAGLE files: [arduino-uno-Rev3-reference-design.zip](#) (NOTE: works with Eagle 6.0 and newer)

Schematic: [arduino-uno-Rev3-schematic.pdf](#)

Note: The Arduino reference design can use an ATmega8, 168, or 328. Current models use an ATmega328, but an ATmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.

- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the [SPI library](#).
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with `analogReference()`.
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega328 ports](#). The mapping for the ATmega8, 168, and 328 is identical.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, [on Windows, a .inf file is required](#). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. For SPI communication, use the [SPI library](#).

Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

B.2 Datasheet PC817

SHARP

PC817 Series

PC817 Series

High Density Mounting Type Photocoupler

- Lead forming type (I type) and taping reel type (P type) are also available. (PC817/PC817P)
- TÜV (VDE0884) approved type is also available as an option.

■ Features

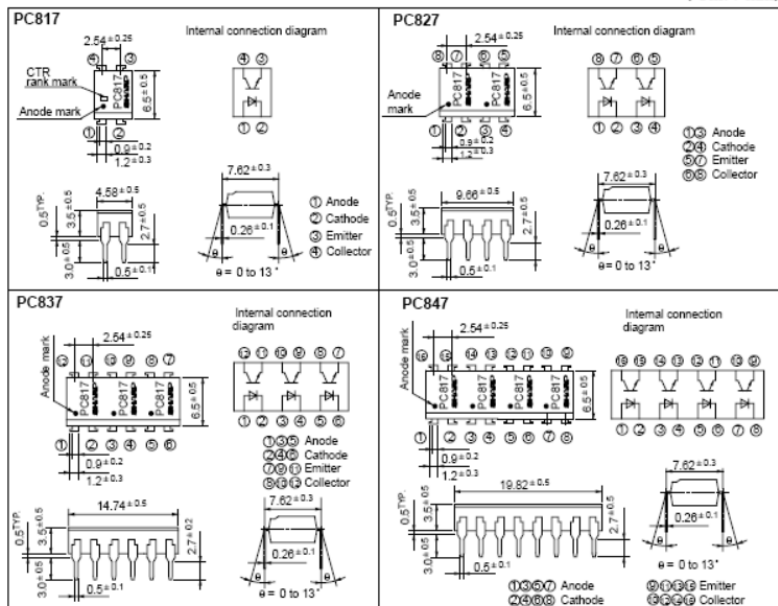
- Current transfer ratio
(CTR: MIN. 50% at $I_F = 5\text{mA}$, $V_{CE} = 5\text{V}$)
- High isolation voltage between input and output (V_{iso} : 5 000V_{rms})
- Compact dual-in-line package
PC817: 1-channel type
PC827: 2-channel type
PC837: 3-channel type
PC847: 4-channel type
- Recognized by UL, file No. E64380

■ Applications

- Computer terminals
- System appliances, measuring instruments
- Registers, copiers, automatic vending machines
- Electric home appliances, such as fan heaters, etc.
- Signal transmission between circuits of different potentials and impedances

■ Outline Dimensions

(Unit: mm)



* In the absence of confirmation by device specification sheets, SHARP takes no responsibility for any defects that occur in equipment using any of SHARP's devices, shown in catalogs, data books, etc. Contact SHARP in order to obtain the latest version of the device specification sheets before using any SHARP's device.*

■ Absolute Maximum Ratings

(Ta = 25°C)

Parameter	Symbol	Rating	Unit
Forward current	I_F	50	mA
*1 Peak forward current	I_{FM}	1	A
Reverse voltage	V_R	6	V
Power dissipation	P	70	mW
Collector-emitter voltage	V_{CEO}	35	V
Emitter-collector voltage	V_{ECO}	6	V
Collector current	I_C	50	mA
Collector power dissipation	P_C	150	mW
Total power dissipation	P_{tot}	200	mW
*2 Isolation voltage	V_{iso}	5 000	V _{rms}
Operating temperature	T_{op}	-30 to +100	°C
Storage temperature	T_{stg}	-55 to +125	°C
*3 Soldering temperature	T_{sol}	260	°C

*1 Pulse width: $\leq 100\mu s$, Duty ratio: 0.001

*2 40 to 60% RH, AC for 1 minute

*3 For 10 seconds

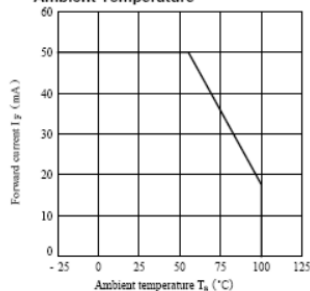
■ Electro-optical Characteristics

(Ta = 25°C)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Forward voltage	V_F	$I_F = 20\text{mA}$	-	1.2	1.4	V
Peak forward voltage	V_{FM}	$I_{FM} = 0.5\text{A}$	-	-	3.0	V
Reverse current	I_R	$V_R = 4\text{V}$	-	-	10	μA
Terminal capacitance	C_t	$V = 0, f = 1\text{kHz}$	-	30	250	pF
Collector dark current	I_{CBO}	$V_{CE} = 20\text{V}$	-	-	10^{-7}	A
*4 Current transfer ratio	CTR	$I_F = 5\text{mA}, V_{CE} = 5\text{V}$	50	-	600	%
Collector-emitter saturation voltage	$V_{CE(sat)}$	$I_F = 20\text{mA}, I_C = 1\text{mA}$	-	0.1	0.2	V
Isolation resistance	R_{iso}	DC500V, 40 to 60% RH	5×10^{10}	10^{11}	-	Ω
Floating capacitance	C_f	$V = 0, f = 1\text{MHz}$	-	0.6	1.0	pF
Cut-off frequency	f_c	$V_{CE} = 5\text{V}, I_C = 2\text{mA}, R_L = 100\Omega, -3\text{dB}$	-	80	-	kHz
Response time	Rise time	$V_{CE} = 2\text{V}, I_C = 2\text{mA}, R_L = 100\Omega$	-	4	18	μs
	Fall time		-	3	18	μs

*4 Classification table of current transfer ratio is shown below.

Fig. 1 Forward Current vs. Ambient Temperature



Model No.	Rank mark	CTR (%)
PC817A	A	80 to 160
PC817B	B	130 to 260
PC817C	C	200 to 400
PC817D	D	300 to 600
PC8●7AB	A or B	80 to 260
PC8●7BC	B or C	130 to 400
PC8●7CD	C or D	200 to 600
PC8●7AC	A, B or C	80 to 400
PC8●7BD	B, C or D	130 to 600
PC8●7AD	A, B, C or D	80 to 600
PC8●7	A, B, C, D or No mark	50 to 600

● : 1 or 2 or 3 or 4

Fig. 2 Collector Power Dissipation vs. Ambient Temperature

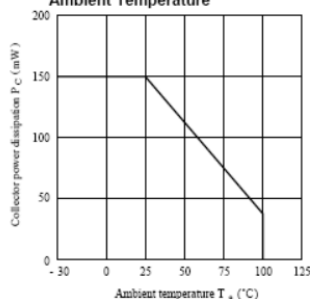


Fig. 3 Peak Forward Current vs. Duty Ratio

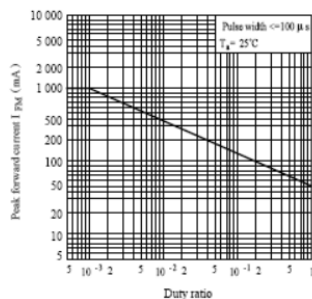


Fig. 4 Current Transfer Ratio vs. Forward Current

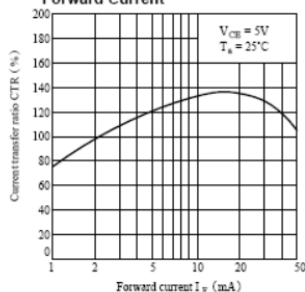


Fig. 5 Forward Current vs. Forward Voltage

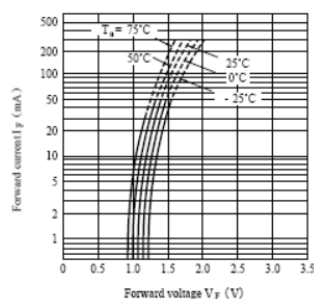


Fig. 6 Collector Current vs. Collector-emitter Voltage

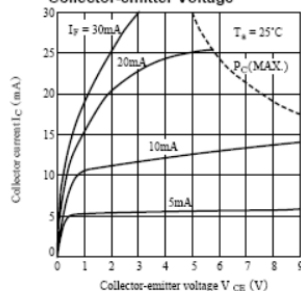


Fig. 7 Relative Current Transfer Ratio vs. Ambient Temperature

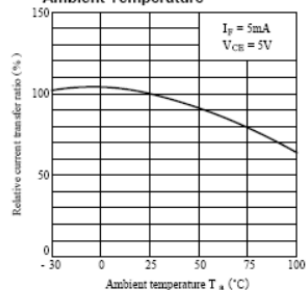


Fig. 8 Collector-emitter Saturation Voltage vs. Ambient Temperature

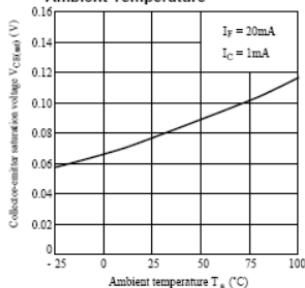


Fig. 9 Collector Dark Current vs. Ambient Temperature

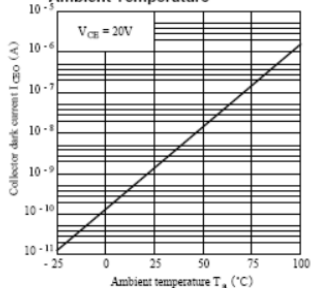


Fig. 10 Response Time vs. Load Resistance

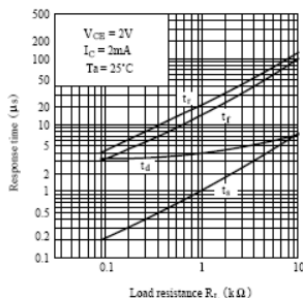
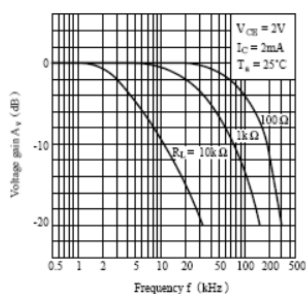
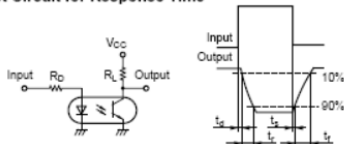


Fig. 11 Frequency Response



Test Circuit for Response Time



Test Circuit for Frequency Response

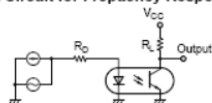
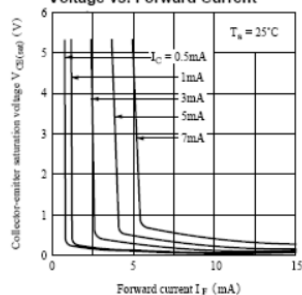


Fig. 12 Collector-emitter Saturation Voltage vs. Forward Current



● Please refer to the chapter "Precautions for Use"

B.3 Datasheet MOC3021

MOTOROLA SEMICONDUCTOR TECHNICAL DATA

Order this document
by MOC3020/D



6-Pin DIP Random-Phase Optoisolators Triac Driver Output (400 Volts Peak)

The MOC3020 Series consists of gallium arsenide infrared emitting diodes, optically coupled to a silicon bilateral switch.

- To order devices that are tested and marked per VDE 0884 requirements, the suffix "V" must be included at end of part number. VDE 0884 is a test option. They are designed for applications requiring isolated triac triggering.

Recommended for 115/240 Vac(rms) Applications:

- Solenoid/Valve Controls
- Lamp Ballasts
- Interfacing Microprocessors to 115 Vac Peripherals
- Motor Controls
- Static ac Power Switch
- Solid State Relays
- Incandescent Lamp Dimmers

MAXIMUM RATINGS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Rating	Symbol	Value	Unit
INFRARED EMITTING DIODE			
Reverse Voltage	V_R	3	Volts
Forward Current — Continuous	I_F	60	mA
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Negligible Power in Triac Driver Derate above 25°C	P_D	100 1.33	mW mW/°C

OUTPUT DRIVER

Off-State Output Terminal Voltage	V_{DRM}	400	Volts
Peak Repetitive Surge Current (PW = 1 ms, 120 pps)	I_{TSM}	1	A
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	300 4	mW mW/°C

TOTAL DEVICE

Isolation Surge Voltage ⁽¹⁾ (Peak ac Voltage, 60 Hz, 1 Second Duration)	V_{ISO}	7500	Vac(pk)
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	330 4.4	mW mW/°C
Junction Temperature Range	T_J	-40 to +100	°C
Ambient Operating Temperature Range ⁽²⁾	T_A	-40 to +85	°C
Storage Temperature Range ⁽²⁾	T_{stg}	-40 to +150	°C
Soldering Temperature (10 s)	T_L	260	°C

- Isolation surge voltage, V_{ISO} , is an internal device dielectric breakdown rating. For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.
- Refer to Quality and Reliability Section in Opto Data Book for information on test conditions.

Preferred devices are Motorola recommended choices for future use and best overall value.
Global Optoisolator is a trademark of Motorola, Inc.

REV 1

© Motorola, Inc. 1995

MOC3021
(IFT = 15 mA Max)
MOC3022
(IFT = 10 mA Max)
MOC3023*
(IFT = 5 mA Max)

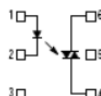
*Motorola Preferred Device

STYLE 6 PLASTIC



STANDARD THRU HOLE
CASE 730A-04

SCHEMATIC



- ANODE
- CATHODE
- NC
- MAIN TERMINAL
- SUBSTRATE
DO NOT CONNECT
- MAIN TERMINAL



MOTOROLA

MOC3021 MOC3022 MOC3023

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
INPUT LED					
Reverse Leakage Current ($V_R = 3\text{ V}$)	I_R	—	0.05	100	μA
Forward Voltage ($I_F = 10\text{ mA}$)	V_F	—	1.15	1.5	Volts
OUTPUT DETECTOR ($I_F = 0$ unless otherwise noted)					
Peak Blocking Current, Either Direction (Rated $V_{DRM}^{(1)}$)	I_{DRM}	—	10	100	nA
Peak On-State Voltage, Either Direction ($I_{TM} = 100\text{ mA Peak}$)	V_{TM}	—	1.8	3	Volts
Critical Rate of Rise of Off-State Voltage (Figure 7, Note 2)	dv/dt	—	10	—	$\text{V}/\mu\text{s}$
COUPLED					
LED Trigger Current, Current Required to Latch Output (Main Terminal Voltage = $3\text{ V}^{(3)}$)	I_{FT}	—	8	15	mA
MOC3021	—	—	—	10	
MOC3022	—	—	—	5	
MOC3023	—	—	—	—	
Holding Current, Either Direction	I_H	—	100	—	μA

- Test voltage must be applied within dv/dt rating.
- This is static dv/dt . See Figure 7 for test circuit. Commutating dv/dt is a function of the load-driving thyristor(s) only.
- All devices are guaranteed to trigger at an I_F value less than or equal to max I_{FT} . Therefore, recommended operating I_F lies between max I_{FT} (15 mA for MOC3021, 10 mA for MOC3022, 5 mA for MOC3023) and absolute max I_F (60 mA).

TYPICAL ELECTRICAL CHARACTERISTICS

$T_A = 25^\circ\text{C}$

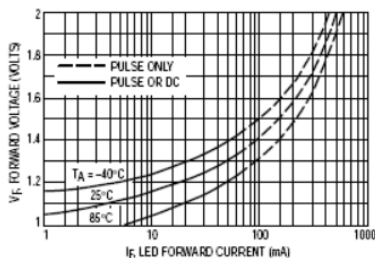


Figure 1. LED Forward Voltage versus Forward Current

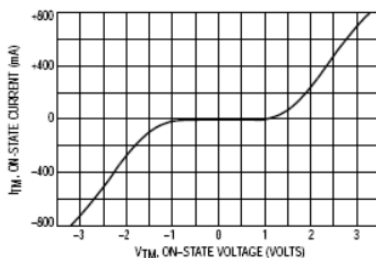


Figure 2. On-State Characteristics

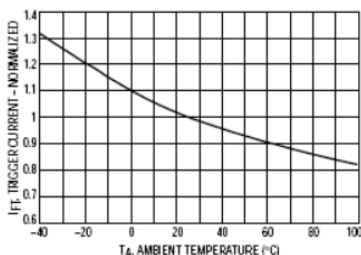


Figure 3. Trigger Current versus Temperature

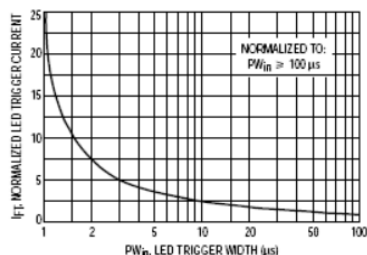


Figure 4. LED Current Required to Trigger versus LED Pulse Width

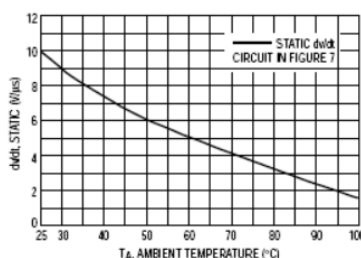


Figure 5. dv/dt versus Temperature

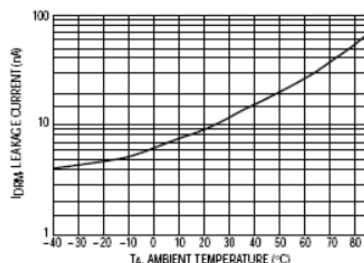
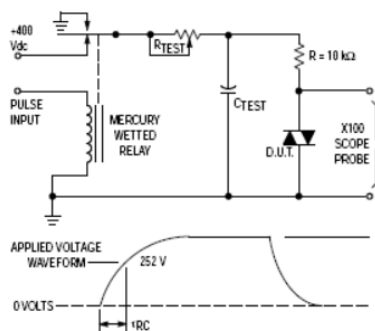
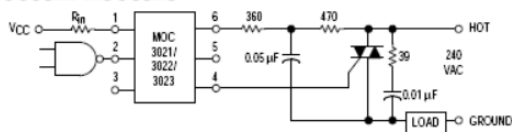

Figure 6. Leakage Current, I_{DRM} versus Temperature


Figure 7. Static dv/dt Test Circuit

1. The mercury wetted relay provides a high speed repeated pulse to the D.U.T.
2. 100x scope probes are used, to allow high speeds and voltages.
3. The worst-case condition for static dv/dt is established by triggering the D.U.T. with a normal LED input current, then removing the current. The variable R_{TEST} allows the dv/dt to be gradually increased until the D.U.T. continues to trigger in response to the applied voltage pulse, even after the LED current has been removed. The dv/dt is then decreased until the D.U.T. stops triggering. 1RC is measured at this point and recorded.

MOC3021 MOC3022 MOC3023



* This optoisolator should not be used to drive a load directly. It is intended to be a trigger device only.

Additional information on the use of optically coupled triac drivers is available in Application Note AN-780A.

In this circuit the "hot" side of the line is switched and the load connected to the cold or ground side.

The 39 ohm resistor and 0.01 μ F capacitor are for snubbing of the triac, and the 470 ohm resistor and 0.05 μ F capacitor are for snubbing the coupler. These components may or may not be necessary depending upon the particular triac and load used.

Figure 8. Typical Application Circuit

B.4 Datasheet BT138

NXP Semiconductors

Product specification

Triacs

BT138 series

GENERAL DESCRIPTION

Passivated triacs in a plastic envelope, intended for use in applications requiring high bidirectional transient and blocking voltage capability and high thermal cycling performance. Typical applications include motor control, industrial and domestic lighting, heating and static switching.

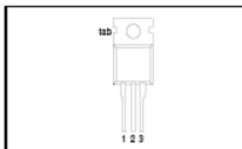
QUICK REFERENCE DATA

SYMBOL	PARAMETER	MAX.	MAX.	UNIT
V_{ORM}	BT138-600F	600	800	V
	BT138-600G	600F	800F	
	BT138-600G	600G	800G	
$I_{T(RMS)}$	RMS on-state current	12	12	A
I_{TSM}	Non-repetitive peak on-state current	95	95	A

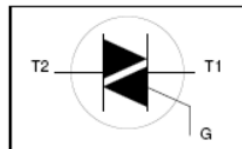
PINNING - TO220AB

PIN	DESCRIPTION
1	main terminal 1
2	main terminal 2
3	gate
tab	main terminal 2

PIN CONFIGURATION



SYMBOL



LIMITING VALUES

Limiting values in accordance with the Absolute Maximum System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.		UNIT
V_{ORM}	Repetitive peak off-state voltages		-	-600 600 ¹	-800 800	V
$I_{T(RMS)}$	RMS on-state current	full sine wave; $T_{MC} \leq 99^\circ\text{C}$	-	12		A
I_{TSM}	Non-repetitive peak on-state current	full sine wave; $T_J = 25^\circ\text{C}$ prior to surge	-	95		A
i^2t	i^2t for fusing	$t = 20\text{ ms}$	-	105		A ² s
di_T/dt	Repetitive rate of rise of on-state current after triggering	$t = 16.7\text{ ms}$	-	45		A/s
		$t = 10\text{ ms}$	-	10		A/s
		$I_{TM} = 20\text{ A}; I_G = 0.2\text{ A}; di_T/dt = 0.2\text{ A}/\mu\text{s}$	-	50		A/ μs
I_{GM}	Peak gate current	T2+ G+	-	50		A/ μs
V_{GM}	Peak gate voltage	T2+ G-	-	50		A/ μs
P_{GM}	Peak gate power	T2- G-	-	10		A/ μs
$P_{GM(AV)}$	Average gate power	T2- G+	-	2		A
T_{JG}	Storage temperature		-	5		V
T_J	Operating junction temperature	over any 20 ms period	-	5		W
			-40	0.5		W
			-	150		$^\circ\text{C}$
			-	125		$^\circ\text{C}$

¹ Although not recommended, off-state voltages up to 800V may be applied without damage, but the triac may switch to the on-state. The rate of rise of current should not exceed 15 A/ μs .

Triacs

BT138 series

THERMAL RESISTANCES

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
R_{th-jb}	Thermal resistance junction to mounting base	full cycle	-	-	1.5	K/W
R_{th-ja}	Thermal resistance junction to ambient	half cycle in free air	-	60	2.0	K/W

STATIC CHARACTERISTICS

 $T_j = 25\text{ }^{\circ}\text{C}$ unless otherwise stated

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.			UNIT
I_{GT}	Gate trigger current	BT138- $V_D = 12\text{ V}; I_T = 0.1\text{ A}$			
		T2+ G+	-	5	35	25	50	mA
		T2+ G-	-	8	35	25	50	mA
		T2- G-	-	10	35	25	50	mA
		T2- G+	-	22	70	70	100	mA
I_L	Latching current	$V_D = 12\text{ V}; I_{GT} = 0.1\text{ A}$						
		T2+ G+	-	7	40	40	60	mA
		T2+ G-	-	20	60	60	90	mA
		T2- G-	-	8	40	40	60	mA
		T2- G+	-	10	60	60	90	mA
I_H	Holding current	$V_D = 12\text{ V}; I_{GT} = 0.1\text{ A}$	-	6	30	30	60	mA
V_T	On-state voltage	$I_T = 15\text{ A}$	-	1.4	1.65			V
V_{GT}	Gate trigger voltage	$V_D = 12\text{ V}; I_T = 0.1\text{ A}$	-	0.7	1.5			V
		$V_D = 400\text{ V}; I_T = 0.1\text{ A};$ $T_j = 125\text{ }^{\circ}\text{C}$	0.25	0.4	-			V
I_D	Off-state leakage current	$V_D = V_{DRM(max)};$ $T_j = 125\text{ }^{\circ}\text{C}$	-	0.1	0.5			mA

DYNAMIC CHARACTERISTICS

 $T_j = 25\text{ }^{\circ}\text{C}$ unless otherwise stated

SYMBOL	PARAMETER	CONDITIONS	MIN.			TYP.	MAX.	UNIT
dV_D/dt	Critical rate of rise of off-state voltage	BT138- $V_{DM} = 67\% V_{DRM(max)};$ $T_j = 125\text{ }^{\circ}\text{C};$ exponential waveform; gate open circuit			
			100	50	200	250	-	V/ μs
dV_{com}/dt	Critical rate of change of commutating voltage	$V_{DM} = 400\text{ V}; T_j = 95\text{ }^{\circ}\text{C};$ $I_{T(RMS)} = 12\text{ A};$ $di_{com}/dt = 5.4\text{ A/ms};$ gate open circuit	-	-	10	20	-	V/ μs
t_{gt}	Gate controlled turn-on time	$I_{HV} = 16\text{ A}; V_D = V_{DRM(max)};$ $I_G = 0.1\text{ A}; di_G/dt = 5\text{ A}/\mu\text{s}$	-	-	-	2	-	μs

Triacs

BT138 series

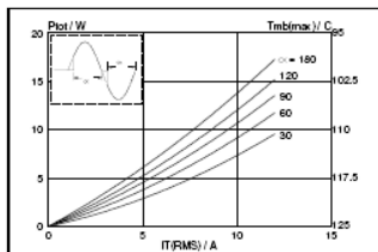


Fig.1. Maximum on-state dissipation, P_{on} , versus rms on-state current, $I_{T(RMS)}$, where α = conduction angle.

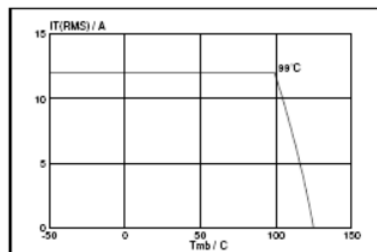


Fig.4. Maximum permissible rms current $I_{T(RMS)}$, versus mounting base temperature T_{mb} .

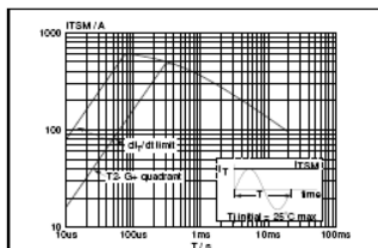


Fig.2. Maximum permissible non-repetitive peak on-state current I_{TSM} versus pulse width t_p for sinusoidal currents, $t_p \leq 20ms$.

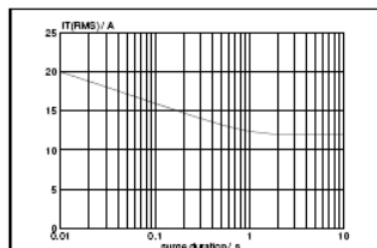


Fig.5. Maximum permissible repetitive rms on-state current $I_{T(RMS)}$, versus surge duration, for sinusoidal currents, $f = 50 Hz$; $T_{amb} \leq 90^\circ C$.

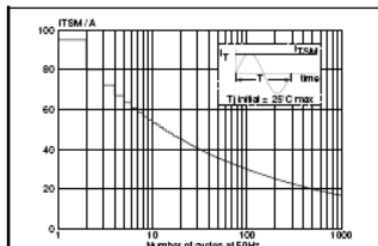


Fig.3. Maximum permissible non-repetitive peak on-state current I_{TSM} versus number of cycles, for sinusoidal currents, $f = 50 Hz$.

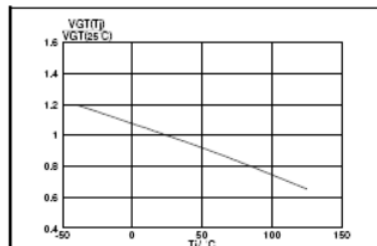


Fig.6. Normalised gate trigger voltage $V_{GT(T)}/V_{GT(25^\circ C)}$, versus junction temperature T_J .

Triacs

BT138 series

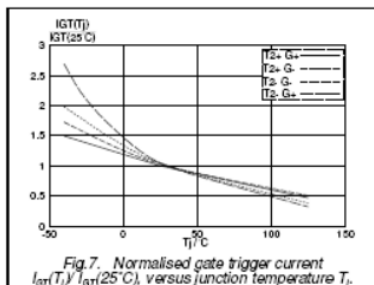


Fig. 7. Normalised gate trigger current $I_{GT}(T_J)/I_{GT}(25^\circ\text{C})$, versus junction temperature T_J .

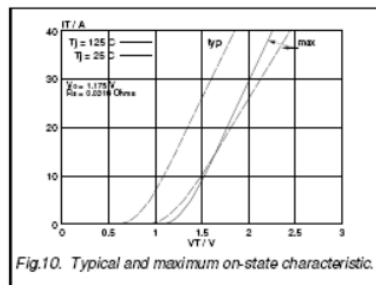


Fig. 10. Typical and maximum on-state characteristic.

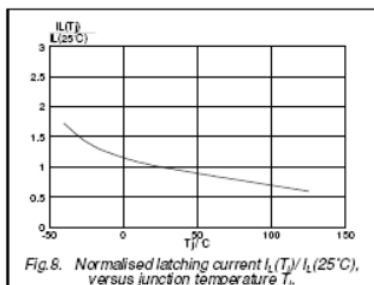


Fig. 8. Normalised latching current $I_L(T_J)/I_L(25^\circ\text{C})$, versus junction temperature T_J .

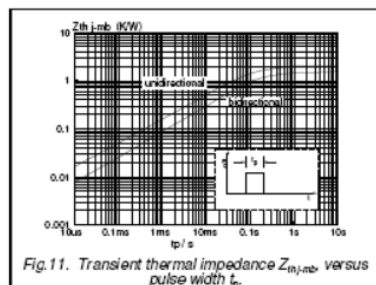


Fig. 11. Transient thermal impedance $Z_{th(j-nb)}$, versus pulse width t_p .

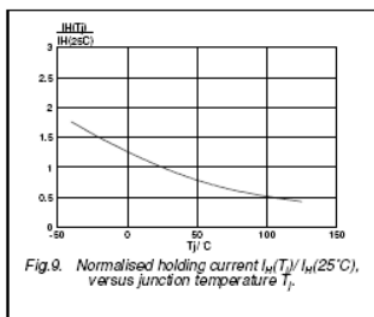


Fig. 9. Normalised holding current $I_H(T_J)/I_H(25^\circ\text{C})$, versus junction temperature T_J .

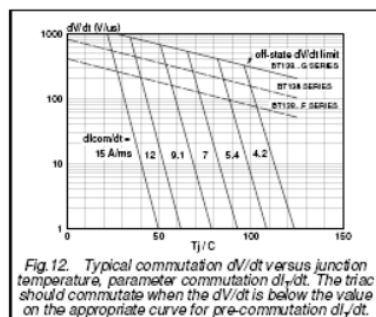


Fig. 12. Typical commutation dV/dt versus junction temperature, parameter commutation dI_T/dt . The triac should commute when the dV/dt is below the value on the appropriate curve for pre-commutation dI_T/dt .

B.5 Datasheet CT103

Mini current transformer CT103



Features:

- 1- High accuracy, AC measurement accuracy up to class 1.0;
- 2- Small phase error, typical phase shift is at 1-10'; Additional compensation not needed;
- 3- Good isolation from Primary to Secondary;
- 4- Very reliable and stable performance at different temperature conditions;
- 5- Frequency: 50Hz/60Hz Ambient Temperature: -40°C ~ 70°C;

Specifications:

- Measurement Current: 100A
- Ratio: Current: 100A/50mA
Turns: 2000
- Load: 12.5Ω
- Insulation Voltage: 3000V
- Phase Error: <15'
- Accuracy Class: 0.1

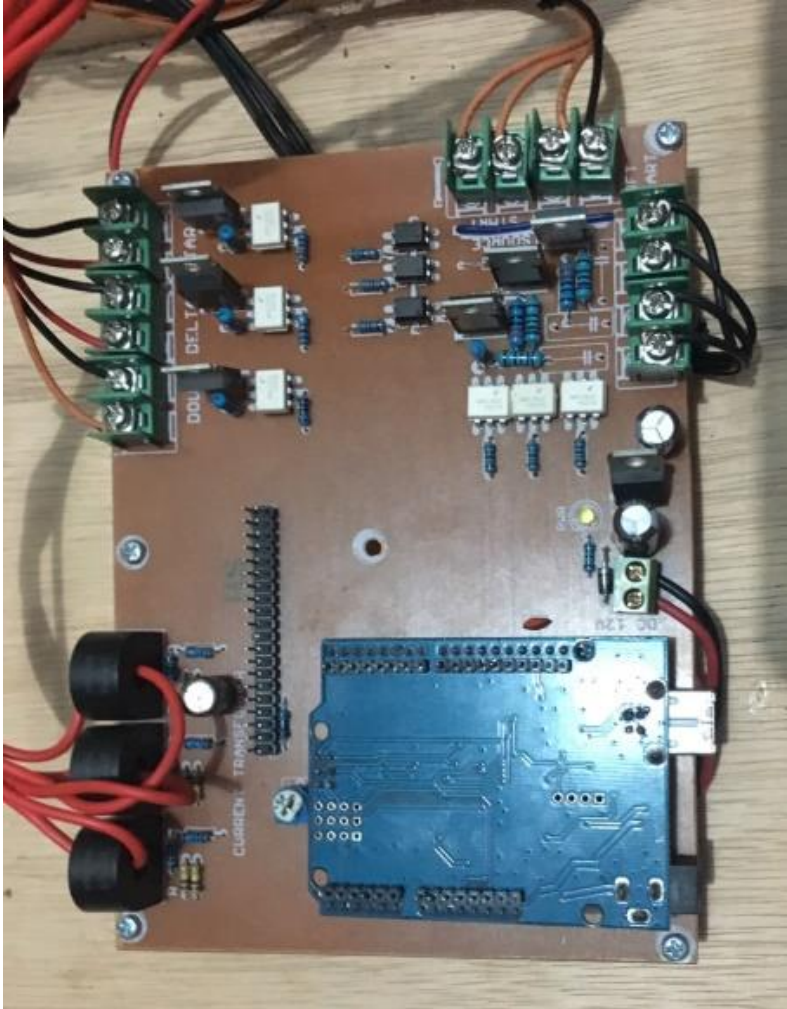
Made in China

EKT
Electronics K&B ang Trading

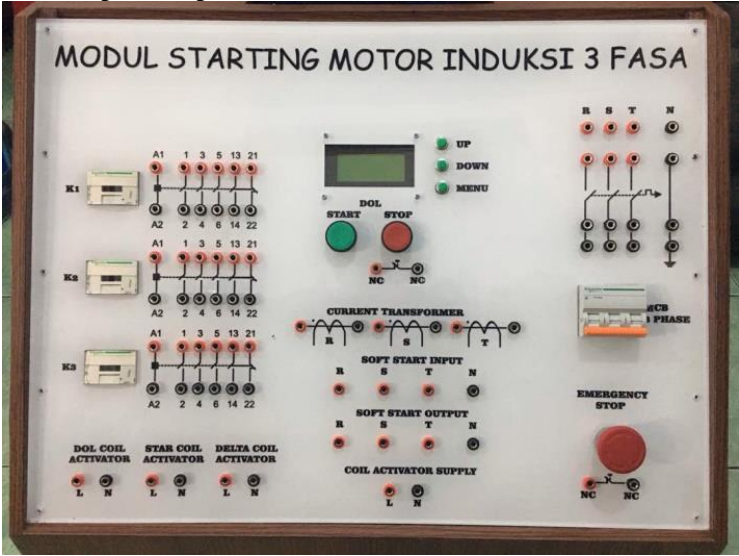
Halaman ini sengaja dikosongi

LAMPIRAN C

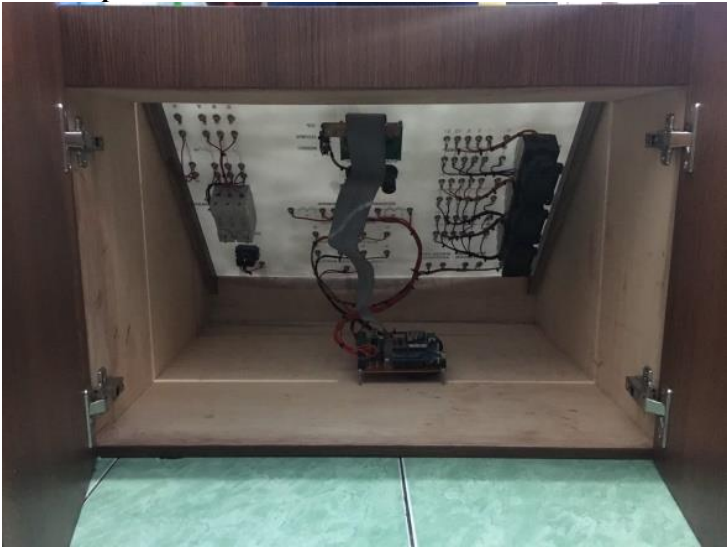
C.1 Tampilan Board



C.2 Tampilan Depan Modul



C.3 Tampilan Didalam Modul



DAFTAR RIWAYAT HIDUP



Nama : Ghiok Nanda Alivsky
TTL : Surabaya, 30 Maret 1996
Jenis Kelamin : Laki - laki
Agama : Islam
Alamat : Gembong II No. 4
Surabaya
Telp/HP : 085736099616
E-mail : ghioknanda@gmail.com

RIWAYAT PENDIDIKAN

1. 2002 – 2008 : SDN Sidotopo Wetan I Surabaya
2. 2008 – 2011 : SMP Negeri 2 Surabaya
3. 2011 – 2014 : SMA Negeri 19 Surabaya
4. 2014 – 2017 : D3 Teknik Elektro Otomasi, Program Studi Teknik Listrik – Fakultas Vokasi Institut Teknologi Sepuluh Nopember (ITS)

PENGALAMAN KERJA

1. Kerja Praktek di PT PLN (Persero) APD Jawa Timur Area Surabaya Selatan.
2. Bartender di Fruts Cafe
3. Receptionist

PENGALAMAN ORGANISASI

1. Presiden Karya Ilmiah Remaja SMAN 19 Surabaya